Robust and Explainable Deep Hedging with Linearized Neural Network

Anonymous authors
Paper under double-blind review

Abstract

Deep hedging is promising for risk management for financial derivatives through deep learning, yet it remains hindered by complex, resource-intensive training and the challenge of effectively integrating deep neural networks with hedging optimization. To overcome these issues, we introduce a robust and efficient linearized neural network architecture, seamlessly integrated with Black-Scholes' Delta, to streamline deep learning-based hedging optimization (DHLNN). Our approach enhances both the efficiency and interpretability of hedging strategies in derivative markets. The proposed model shows strong resilience to market fluctuations, effectively addresses action-dependence challenges, and achieves faster convergence compared to existing methods. Extensive simulations confirm the superior performance and cost-effectiveness of our method, under varying market conditions, when compared to state-of-the-art deep hedging models. These findings underscore the potential of DHLNN to significantly improve both convergence and hedging performance in derivative markets.

1 Introduction

The evolution of trading practices has transcended traditional, model-driven pricing and hedging techniques. In the dynamic landscape of contemporary financial markets, a more sophisticated, data-driven approach has become indispensable. Deep neural networks (DNNs) have emerged as powerful tools in this context, offering unprecedented representational power. By harnessing the strengths of DNNs alongside advancements in modern optimization algorithms, we can advance beyond the conventional reliance on Greeks as the primary risk management tool, paving the way for more robust and promising hedging strategies Buehler et al. (2019).

Deep hedging has emerged as the state-of-the-art framework for automating hedging operations Buehler et al. (2019), offering effective risk mitigation with an inherent consideration for path dependence. Notably, the No-Transaction Band network (NTB) has been proposed as a rapid method for determining optimal hedging strategies, enabling hedgers to respond swiftly to customer demands by providing faster and more accurate quotes Imaki et al. (2021). However, translating the deep hedging concept into practical application presents significant challenges.

One of the most critical challenges in deep hedging is action-dependence, where the optimal hedging position at a subsequent time step is influenced by the current action. This dependency forces the neural network to navigate an expansive function space, complicating the optimization process Kallsen & Muhle-Karbe (2015). Furthermore, the non-convex nature of the loss surface in neural networks, riddled with saddle points, intensifies this situation, often leading to convergence issues despite numerous gradient updates Karakida et al. (2019). For hedgers who must execute orders timely and accurately, this lack of convergence poses a significant risk.

In response to these challenges, several directions to enhance the performance of deep hedging could come from several recent studies in machine learning. Analyzing the training dynamics of wide-width shallow networks has provided valuable insights into the optimization behavior of neural networks Mei et al. (2018); Novak et al. (2022). Moreover, task-specific design integration into neural networks has frequently led to substantial advancements, as evidenced by breakthroughs in various deep learning applications Krizhevsky

et al. (2012); Vaswani et al. (2017). These approaches underscore the potential for enhancing the convergence and stability of deep hedging models. Nevertheless, the landscape of deep hedging remains fraught with complexities. The inclusion of transaction costs, for instance, adds another layer of difficulty. Hedging in the presence of market frictions, such as transaction costs, necessitates sophisticated adjustments to the hedging strategy, which may not be adequately captured by conventional deep learning models. Additionally, ensuring that these models remain robust and reliable in dynamic and unpredictable market conditions remains an ongoing challenge.

To address these challenges, we propose a novel approach that integrates a linearized neural network architecture with the well-established Black-Scholes' Delta. This design mitigates the complexities associated with action-dependence by making hedge positions a function of current market conditions rather than the entire history of previous actions. As a result, our model can efficiently manage high-volume orders, enabling rapid and precise adjustments to hedge positions in response to market changes. This capability is particularly critical in volatile markets or when handling large transaction volumes, where delays in decision-making can lead to significant losses. Moreover, our approach introduces an optimization strategy that spans the entire price trajectory of the underlying asset from the inception of the derivative to its maturity. This holistic method ensures that the sequence of hedge positions is optimized as a whole, capturing the full dynamics of the hedging process. By optimizing across the entire trajectory, our model delivers more consistent and effective management of derivative positions, ensuring alignment with the overall objective of minimizing risk or cost throughout the life of the derivative. In summary, the contributions of this work are as follows:

- We propose a novel framework based on neural network linearization that addresses the core challenges of deep hedging. This framework offers a robust and interpretable solution by blending the strengths of neural networks with the clarity of linear models.
- Our approach introduces an optimization strategy that spans the entire price trajectory, ensuring a holistic and effective method for managing derivative positions. This strategy is crucial for accurately capturing the dynamics of hedging over time.
- By integrating the linearized neural network architecture with Black-Scholes' Delta, we overcome the challenge of action-dependence. This integration facilitates the management of high-volume orders and the rapid injection of liquidity into derivative markets, thereby enhancing overall market efficiency.
- Extensive simulations demonstrate the resilience of our proposed approach under varying market conditions. The results highlight the model's superior convergence speed and robust performance, making it a more resilient and efficient approach to risk management in derivative markets.

The rest of this paper is organized as follows. Section 2 introduces the background and related work. Section 3 formulates the hedging optimization problem, followed by Section 4 which explicates the proposed method. Section 6 shows the results where the underlying asset price follows geometric Brownian motion. The concluding remarks are given in Section 7.

2 Related Work

Derivatives are financial instruments whose value is derived from underlying assets such as equities, bonds, and currencies. Frequent trading of derivatives often leads to the initiation of short positions, which necessitates sophisticated hedging strategies to mitigate the associated risks. The value of a derivative closely tracks the fluctuations in the underlying asset, exhibiting a level of sensitivity that necessitates strategic interventions by hedgers. They manage this variability by executing transactions that precisely adjust their exposure to the underlying asset, thereby offsetting risk. This delicate balance between derivatives and their underlying assets emphasizes the intricate mechanics necessary for effective hedging within the complex landscape of financial markets.

In a frictionless and fully efficient market, perfect hedging can theoretically be achieved by employing the Black-Scholes replicating portfolio Black & Scholes (1973). However, the Black-Scholes framework

assumes that derivatives can be perfectly replicated using underlying instruments, effectively rendering them redundant in such an idealized market Davis et al. (1993). In contrast, real-world markets are characterized by various frictions, such as transaction costs, which significantly complicate the optimization of hedging strategies Collins & Fabozzi (1991). Extending the Black-Scholes model to account for transaction costs has led to the development of discretely rebalanced portfolios Leland (1985), as well as approaches that approximate the optimal hedge in the mean-variance sense Grannan & Swindle (1996). Additionally, utility indifference pricing has been employed in several works to derive optimal hedging strategies Monoyios (2004)Henderson & Hobson (2004)Carmona (2008). Despite these advancements, achieving a truly optimal hedge that fully accounts for transaction costs remains an unresolved challenge.

While there is extensive literature on pricing models employing neural networks, only a limited number of studies have explored the application of neural networks to hedging strategies Ruf & Wang (2019). Buehler et al. (2019) demonstrated that deep hedging algorithms could effectively hedge a European option in the presence of market frictions, using convex risk measures as the loss function. Imajo et al. (2021) proposed a neural network architecture that incorporates widely accepted financial inductive biases, such as the amplitude and time-scale variance of stock time series, to enhance performance. Furthermore, hybrid models combining parametric option pricing approaches like the Black-Scholes model with non-parametric machine learning techniques have been proposed to improve predictive accuracy by incorporating inductive biases that categorize options based on moneyness and time-to-maturity Das & Padhy (2017). Jang & Lee (2019) introduced generative Bayesian neural networks, consistent with the no-arbitrage pricing structure, to achieve better calibration and prediction performance for American option pricing compared to classical models. Neural networks have also been applied to analyze volatility surface movements, offering deeper insights into market dynamics Cao et al. (2020).

In response to the unresolved challenges in deep hedging, we introduce an efficient method that enhances training by fully leveraging the versatile representation capabilities of neural network models, while simultaneously harnessing the explainability of linear models as an approximation of the neural network. Our approach addresses the complexities of action-dependence, demonstrates resilience to varying market conditions, and exhibits superior convergence speed. By focusing on the linearization of neural network models and integrating them with deep hedging strategies, the proposed DHLNN model offers a more efficient and interpretable approach to optimizing hedge strategies in derivative markets. Not only does DHLNN outperform traditional hedging methods, but it also surpasses state-of-the-art deep hedging baselines, particularly in volatile market conditions, making it a valuable tool for traders and financial institutions aiming to enhance their hedging strategies and reduce risk.

3 Hedging Optimization

In this section, we formulate the optimal hedging problem in three steps. First, we describe the derivative market considering transaction costs. Next, we construct the optimal derivative hedging objective, which aims to mitigate liabilities by minimizing the convex risk measure. Finally, we formulate the optimal hedging problem in terms of the certainty equivalent of a convex risk measure, specifically utilizing the exponential utility framework within the context of indifference pricing.

The payoff of a derivative depends on the price trajectory of the underlying asset Hull & White (1987). When hedgers enter into derivative contracts, they commit to settling the resulting payoff at maturity. This commitment involves the obligation to either buy or sell the underlying asset to the derivative holder on the agreed-upon date at the specified price. The pricing of a derivative contract is intricately connected to the price movements of the underlying asset, providing hedgers with crucial insights for determining their optimal position.

Consequently, hedgers expose themselves to potentially higher risks, as they must fulfill their liabilities when the derivatives are exercised. This can result in losses exceeding the initial options premium paid. Therefore, careful consideration of the underlying asset's price dynamics is imperative for effective risk management in derivative trading. The objective of hedging optimization is to design the hedging positions over the underlying asset price trajectories to minimize the capital required to be injected into the portfolio to balance their liabilities.

3.1 The Derivative Market Formulation

In financial markets, a sole tradable asset is often represented by its Weighted Average Price (WAP), which provides a comprehensive measure of its value. The order book, which lists all outstanding buy (bid) and sell (ask) orders, is the primary source of information for calculating the WAP. We examine the market characterized by discrete time steps

$$t = \{t_0 = 0, t_1, t_2, \cdots, t_n = T\}. \tag{1}$$

The WAP at a given time t_i, denoted by P_{t_i}, incorporates both the prices and the sizes of the orders. To calculate the WAP, we first determine the weighted average bid price and the weighted average ask price.

The weighted average bid price (P_{bid,t_i}) is given by

$$P_{\text{bid},t_i} = \frac{\sum_j (\text{Bid Price}_j \times \text{Bid Size}_j)}{\sum_j \text{Bid Size}_j}, \tag{2}$$

where j indexes the different bid prices and sizes. Similarly, the weighted average ask price (P_{ask,t_i}) is calculated as

$$P_{\text{ask},t_i} = \frac{\sum_k (\text{Ask Price}_k \times \text{Ask Size}_k)}{\sum_k \text{Ask Size}_k}, \tag{3}$$

where k indexes the different ask prices and sizes.

The WAP (P_{t_i}) is then determined by averaging the weighted average bid and ask prices as

$$P_{t_i} = \frac{P_{\text{bid},t_i} + P_{\text{ask},t_i}}{2}. \tag{4}$$

Over a period from $0 \leq t_i \leq T$, the series of WAPs for a sole tradable asset is represented as

$$P = \{P_{t_i} | P_{t_i} > 0\}_{0 \leq t_i \leq T}. \tag{5}$$

The WAP provides a fairer valuation of the asset by incorporating the sizes of the orders, which is crucial for making informed trading decisions. It offers a clearer picture of market liquidity, helping traders understand the market's depth and the potential impact of large orders.

When dealing with financial derivatives, it is important to specify how and when the payoff is settled. In our model, we assume that the payoff is settled at maturity $t = T$ and discounted at the zero risk-free rate. Maturity refers to the specific date on which the derivative contract expires. At this point, the obligations of the contract are fulfilled. Settlement at maturity means that any profits or losses from the derivative are realized at this end date. For example, in the case of a European call option, the holder has the right to buy the underlying asset at the strike price only on the maturity date T.

Discounting is a financial process where future cash flows are adjusted to their present value. This is essential because a dollar today is worth more than a dollar tomorrow due to the potential earning capacity of the money. Discounting at the zero risk-free rate implies that we are using this rate to calculate the present value of the derivative's payoff at maturity as shown in the process

$$P_v = \frac{P_T}{(1+r)^T}, \tag{6}$$

where P_v is the present value, P_T is the future value (payoff at maturity), r is the risk-free rate, and T is the time until maturity. Hence, under the assumption of a zero risk-free rate, the payoff at maturity does not need to be adjusted for present value, simplifying our calculations.

The liabilities of the hedger holding a European call option can be expressed as

$$Z = \max(P_T - P_s, 0), \tag{7}$$

where P_T is the price of the asset at maturity T and P_s is the strike price. The liabilities of a Lookback call option can be expressed as

$$Z = \max\left(\max\{P_{t_i} \mid P_{t_i} > 0\}_{0 \le t_i \le T} - P_s, 0\right), \tag{8}$$

where the maximum price $\max\{P_{t_i} \mid P_{t_i} > 0\}_{0 \le t_i \le T}$ is observed over the entire time period from t_0 to T.

Mitigating investment risk involves strategically employing financial instruments or market strategies to counteract potential adverse price movements. Hedgers manage the risk associated with their liabilities by trading the underlying asset of the derivative. The position of the underlying asset held by the hedger at each time step is indicated by a signed numerical quantity denoted as

$$\delta = \{\delta_{t_i} \mid \delta_{t_i} \in \mathbb{R}\}_{0 \le t_i \le T}, \tag{9}$$

where $\delta_0 = 0$. The hedgers need to determine δ_{t_i} based on the information available before t_i. The market imposes a transaction cost that is proportional to traded values, characterized by a cost rate $c \in [0, 1)$. The total transaction costs incurred by the hedger until t_{i+1} are denoted by

$$C_{t_{i+1}} = C_{t_i} + cP_{t_i}|\delta_{t_{i+1}} - \delta_{t_i}|, \tag{10}$$

where $C_0 = 0$. In practical scenarios, transaction costs comprise a range of factors, including the bid-ask spread, commission fees, temporary market impact, and other associated expenses.

3.2 Derivative Hedging Objective

Investors strategically hedge one investment by executing trades in another. The overall objective of investment hedging is to mitigate the impact of adverse events. For derivative instruments, hedgers employ a trading strategy denoted as $\boldsymbol{\delta}$ for the underlying assets to offset the associated liabilities.

All trading operations are self-financed, the liabilities of the hedger can be reflected by the amount of injecting additional cash P_L into the portfolio. It is noteworthy that a negative cash injection implies the potential for the hedger to extract cash which means the positive terminal portfolio value. Importantly, the primary objective is not terminal portfolio value maximization over the hedging strategy. Instead, the ultimate goal is to hedge the liability Z at T through strategic trading in the underlying asset, where the hedging objective should adhere to:

$$P_L - Z + \sum_{t_i=0}^{T-1} \delta_{t_i} \Delta P_{t_i} - C_{t_i}|\delta_{t_{i+1}} - \delta_{t_i}|P_{t_i} = 0. \tag{11}$$

We consider the mid-price of the underline asset denoted by $P = \{P_{t_i}|P_{t_i} > 0\}_{0 \le t_i \le T}$.

The hedger must establish an optimality criterion for the hedging strategy to provide the least amount of cash required to supplement the liabilities, implementing the optimal hedge accounting for various costs and constraints. We formulate criteria to describe the target of hedging optimization, employing convex risk measures defined as

$$\min_{\delta} \quad \rho(-Z + \sum_{t_i=0}^{T-1} \delta_{t_i} \Delta P_{t_i} - C_{t_i}|\delta_{t_{i+1}} - \delta_{t_i}|P_{t_i}), \tag{12}$$

where ρ is a monotonically decreasing convex function with Cash-Invariant property Ben-Tal & Teboulle (2007). The solution to (12) is referred to as the optimal hedging strategy for position $-Z$. This strategy determines the minimal capital required to be added to the risky position, rendering it acceptable according to the risk measure. In essence, hedging optimization aims to find the minimum charge that the hedger needs to implement in order to make the terminal position acceptable when employing optimal hedging.

3.3 Certainty Equivalent of Convex Risk Measures in Indifference Pricing

To achieve indifference between the position $-Z$ and the scenario without liability, which represents the acceptable case with no capital injection, we introduce the indifference price $q(Z)$ as:

$$\begin{aligned} q(Z) = &\inf_{\delta} \rho\Big(-Z + \sum_{t_i=0}^{T-1} \delta_{t_i} \Delta P_{t_i} - C_{t_i}|\delta_{t_{i+1}} - \delta_{t_i}|P_{t_i}\Big) \\ &- \inf_{\delta} \rho\Big(\sum_{t_i=0}^{T-1} \delta_{t_i} \Delta P_{t_i} - C_{t_i}|\delta_{t_{i+1}} - \delta_{t_i}|P_{t_i}\Big). \end{aligned} \tag{13}$$

Indifference pricing provides a natural and versatile framework for characterizing the optimality of hedging strategies and determining the fair price of derivatives in the presence of market frictions. The indifference price $q(Z)$ can be interpreted as the amount of capital needed to injected into the portfolio to eliminate the liability Z, which means inject $q(Z)$ can achieve the same expected utility as the case without liability.

The optimal hedging and pricing problem is converted into an optimization problem of the convex risk measure. According to the exponential utility $U(x) = -\exp(-\lambda x)$, we define a continuous, non-decreasing, and convex loss function L as

$$L(x) = e^{\lambda x} - \frac{1 + \log(\lambda)}{\lambda}, \tag{14}$$

where $\lambda > 0$ is a risk aversion coefficient. Then, we establish an optimized certainty equivalent of the convex risk measure. This measure can be formulated as an optimization problem, aiming to minimize the certainty equivalent under the given loss function L

$$\rho_Z = \inf_{\theta \in \mathbb{R}} \Big\{ \theta + \mathbb{E}[L(Z - \theta - \sum_{t_i=0}^{T-1} (\delta_{t_i} \Delta P_{t_i} - C_{t_i}|\delta_{t_{i+1}} - \delta_{t_i}|P_{t_i}))] \Big\}, \tag{15}$$

where θ is an optimization variable. Substituting the loss function $L(x) = e^{\lambda x} - \frac{1+\log(\lambda)}{\lambda}$ into the optimization problem, we get

$$\rho_Z = \inf_{\theta \in \mathbb{R}} \Big\{ \theta + \mathbb{E}[e^{\lambda(Z - \theta - \sum_{t_i=0}^{T-1} (\delta_{t_i} \Delta P_{t_i} - C_{t_i}|\delta_{t_{i+1}} - \delta_{t_i}|P_{t_i}))}] - \frac{1 + \log(\lambda)}{\lambda} \Big\}. \tag{16}$$

We denote X as follows

$$X = Z - \sum_{t_i=0}^{T-1} (\delta_{t_i} \Delta P_{t_i} - C_{t_i}|\delta_{t_{i+1}} - \delta_{t_i}|P_{t_i}), \tag{17}$$

which simplifies our problem to

$$\rho_Z = \inf_{\theta \in \mathbb{R}} \Big\{ \theta + \mathbb{E}[e^{\lambda(X - \theta)}] - \frac{1 + \log(\lambda)}{\lambda} \Big\}. \tag{18}$$

We solve the optimization problem by differentiating with respect to θ and setting the derivative to zero to find the optimal θ

$$\frac{\partial}{\partial \theta} \Big\{ \theta + \mathbb{E}[e^{\lambda(X - \theta)}] - \frac{1 + \log(\lambda)}{\lambda} \Big\} = 1 - \lambda \mathbb{E}[e^{\lambda(X - \theta)}] = 0. \tag{19}$$

Thus, we have

$$\lambda \mathbb{E}[e^{\lambda(X - \theta)}] = 1 \quad \Rightarrow \quad \mathbb{E}[e^{\lambda(X - \theta)}] = \frac{1}{\lambda}. \tag{20}$$

Taking the logarithm on both sides, we get

$$\lambda(X - \theta) = \log\left(\frac{1}{\lambda}\right) = -\log(\lambda) \quad \Rightarrow \quad X - \theta = -\frac{\log(\lambda)}{\lambda}, \tag{21}$$

which leads to

$$\theta = X + \frac{\log(\lambda)}{\lambda}. \tag{22}$$

Substitute the optimal θ back into the original optimization problem, we obtain the entropic risk measure as

$$\rho_Z = \frac{1}{\lambda}\log(\lambda\mathbb{E}[e^{\lambda(Z-\sum_{t_i=0}^{T-1}(\delta_{t_i}\Delta P_{t_i}-C_{t_i}|\delta_{t_{i+1}}-\delta_{t_i}|P_{t_i}))}]). \tag{23}$$

This completes the detailed derivation of converting the optimal hedging and pricing problem into an optimization problem using a convex risk measure with an exponential utility function.

The fair price $q(Z)$ corresponds to the cash amount that renders a hedger the equivalent situation of settling a liability with $q(Z)$ and having no liability, assuming optimal hedging. The optimal convex risk measure is construed as the residual risk of the derivative post-optimal hedging. Building on the foundation of the optimal hedge within the framework of the hedger's risk measure, the hedger quotes a price that compensates for the remaining risk after hedging.

4 Deep Hedging with Linearized Neural Network

In addressing challenges related to convergence difficulties and non-explanatory training dynamics in neural network hedging, we introduce an innovative approach known as deep hedging with a linearized neural network (DHLNN). This strategy involves leveraging the outputs of a linearized neural network alongside Black-Scholes' delta to formulate an effective hedging strategy.

The key innovation lies in transforming the complex, infinite-dimensional task of identifying an optimal hedging strategy into a more manageable, finite-dimensional challenge. Specifically, our focus is on determining optimal parameters for the linearized neural network, streamlining the overall optimization process. Additionally, the integration of Black-Scholes' delta enhances efficiency and holds the potential for substantial reductions in transaction costs.

A significant advantage of employing a linearized neural network is its enhanced robustness in dynamic market conditions. This not only streamlines the hedging optimization process but also provides a clearer and more interpretable understanding of the training dynamics. This approach aligns well with the practicalities of financial markets, making it a valuable tool for risk management.

In this section, first, we formulate the neural model architecture. Second, we design the hedging model with robust and explainable properties. Finally, we provide the analysis of the model approximation by linearization.

4.1 Market Information

In structuring the market information, we incorporate three crucial factors for derivative pricing and hedging: the log moneyness of the underlying asset, the expiration time, and the volatility of the underlying asset. These factors provide a comprehensive representation of the market information relevant to derivative valuation and hedging.

Log moneyness is a measure that compares the current price of the underlying asset P_{t_i} to the strike price P_s of the derivative. It is calculated as the natural logarithm of the ratio between these two prices, $\log\left(\frac{P_{t_i}}{P_s}\right)$. This measure indicates whether an option is in-the-money, at-the-money, or out-of-the-money. In-the-money means the option would have a positive payoff if exercised today; at-the-money means the current price is roughly equal to the strike price; and out-of-the-money means it would not be profitable to exercise the option at present. Using the logarithm standardizes this measure and makes it symmetric for different strike prices, which aids in neural network training by mitigating issues related to scale differences.

The expiration time $T - t_i$ represents the remaining time until the derivative contract reaches its maturity T. This factor is critical because the value of a derivative, particularly options, is highly sensitive to time. As the expiration date approaches, the time value of the option decreases. The closer the expiration date, the

lesser the impact of potential future price movements on the derivative's value, making it an essential input for pricing models.

Volatility, denoted by σ, measures the degree of variation of the underlying asset's price over time. It reflects the market's expectation of the asset's price fluctuation and is a fundamental input in the valuation of derivatives. Higher volatility increases the potential for the underlying asset's price to move significantly, thus increasing the value of options. Conversely, lower volatility implies less price movement, reducing the option's value. For neural networks, volatility provides a critical measure of risk and price dynamics, influencing the model's ability to predict and hedge the value of derivatives accurately.

While these three factors are not the only market information available, they are representative and encompass much of the essential data needed for derivative pricing and hedging. Many other features can be derived from these factors to represent different aspects of market information. By including log moneyness, expiration time, and volatility, our neural network model captures the fundamental elements affecting the pricing and hedging of derivatives. This comprehensive approach ensures that the model is well-equipped to handle the complexities of financial markets, providing accurate and reliable outputs for risk management and trading strategies.

4.2 Model Function Formulation

The neural network processes market information as inputs $\boldsymbol{x}_{t_i}$ and produces a permissible band for the subsequent position, effectively addressing the issue of position-dependence. The parameters of the neural network model are represented by the weight matrices and bias vectors across multiple layers. Let $\boldsymbol{W}^{(l)}$ and $\boldsymbol{b}^{(l)}$ denote the weights and biases of the l-th layer, where $l = 1, \ldots, L-1$. The hidden layers are indexed by $l = 1, \ldots, L-1$. The parameters of the neural network model are represented by $\boldsymbol{w} = \{\boldsymbol{W^{(l)}}, \boldsymbol{b^{(l)}}\}_{l=1}^{L-1}$ and $\boldsymbol{a_1}, \boldsymbol{a_2} \in \mathbb{R}^m$.

Given an input $\boldsymbol{x}_{t_i}$, the neural network processes this input through $L-1$ hidden layers. The transformation starts with the first hidden layer, which computes

$$\boldsymbol{h}^{(1)} = \sigma(\boldsymbol{W}^{(1)}\boldsymbol{x}_{t_i} + \boldsymbol{b}^{(1)}). \tag{24}$$

Subsequent hidden layers apply the transformation

$$\boldsymbol{h}^{(l)} = \sigma(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \quad \text{for} \quad l = 2, \ldots, L-1. \tag{25}$$

Finally, the output layer applies a linear transformation to the outputs of the last hidden layer. This transformation is weighted by the vectors $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ to produce the two outputs $f_1(\boldsymbol{x}_{t_i})$ and $f_2(\boldsymbol{x}_{t_i})$. The model function, denoted as $f : \mathbb{R}^d \to \mathbb{R}^2$ with L hidden layers comprising is defined as

$$\left\{\boldsymbol{f}_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) = \frac{1}{\sqrt{m}} \sum_{j=1}^{m} \boldsymbol{a}_{k,j} \prod_{l=1}^{L} \left[\sigma(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)})\right]\right\}_{k=1,2}. \tag{26}$$

The neural network representation facilitates the translation of the hedging problem into the optimization of a set of parameters $\boldsymbol{w} = \{\boldsymbol{W^{(l)}}, \boldsymbol{b^{(l)}}\}_{l=1}^{L-1}$ of the neural network, guided by the convex risk measure in (23). The minimizer of (23) provides an approximate value for the indifference price, as demonstrated in (13). This comprehensive methodology offers a promising avenue for addressing challenges in neural network hedging within the context of financial markets.

The neural network's design effectively tackles the challenge of position-dependence, which refers to the difficulty of making predictions or decisions that are influenced by the specific position or state of a financial instrument. By using this approach, the model is able to incorporate various market factors and dependencies into its predictions, providing a flexible and adaptive mechanism for financial decision-making. This capability is crucial in financial contexts where market conditions and positions can vary widely, ensuring that the network's outputs remain relevant and accurate across different scenarios.

4.3 Robust and Explainable Hedging Model Design

Designing a robust and explainable hedging model involves developing a strategy that not only effectively mitigates risks but also provides clear insights into its decision-making process. To make the model more robust against the dynamic market information and explainable about the training dynamics, we introduce the linearized neural network coupling with Black-Scholes' delta to formulate an effective hedging strategy.

4.3.1 Training Dynamic

We establish the dynamics of the model weights throughout the updating trajectory. In order to explicate the training dynamics of the transition from the current model $\boldsymbol{w^t}$ to a new model $\boldsymbol{w}$, we introduce an auxiliary variable denoted as $\boldsymbol{z}$:

$$\boldsymbol{z} = \boldsymbol{w^t} + \tau(\boldsymbol{w} - \boldsymbol{w^t}) \tag{27}$$

where $0 \leq \tau \leq 1$. With $\boldsymbol{z}$ representing the model parameters, we obtain the first-order derivative of the model function as:

$$\left\{ \frac{d\boldsymbol{f}_k(\boldsymbol{x}_{t_i}, \boldsymbol{z}, \boldsymbol{a}_k)}{d\tau} = \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{z}, \boldsymbol{a_k})^{\boldsymbol{T}}(\boldsymbol{w} - \boldsymbol{w^t}) \right\}_{k=1,2}, \tag{28}$$

which leads to

$$\left\{ \int_0^1 d\boldsymbol{f}_k(\boldsymbol{x}_{t_i}, \boldsymbol{z}, \boldsymbol{a}_k) = \int_0^1 \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{z}, \boldsymbol{a_k})^{\boldsymbol{T}}(\boldsymbol{w} - \boldsymbol{w^t}) d\tau \right\}_{k=1,2}. \tag{29}$$

According to

$$\left\{ \boldsymbol{f}_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) - \boldsymbol{f}_k(\boldsymbol{x}_{t_i}, \boldsymbol{w^t}, \boldsymbol{a}_k) = \int_0^1 f_k(\boldsymbol{w^t} + \boldsymbol{\tau}(\boldsymbol{w} - \boldsymbol{w^t})) d\tau \right\}_{k=1,2}, \tag{30}$$

and combining with (29), we have

$$\left\{ \boldsymbol{f}_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) - \boldsymbol{f}_k(\boldsymbol{x}_{t_i}, \boldsymbol{w^t}, \boldsymbol{a}_k) = \int_0^1 \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{z}, \boldsymbol{a_k})^{\boldsymbol{T}}(\boldsymbol{w} - \boldsymbol{w^t}) d\tau \right\}_{k=1,2}, \tag{31}$$

which can be rewritten as

$$\begin{aligned} &\Big\{ \boldsymbol{f}_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) - \boldsymbol{f}_k(\boldsymbol{x}_{t_i}, \boldsymbol{w^t}, \boldsymbol{a}_k) = \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k})^{\boldsymbol{T}}(\boldsymbol{w} - \boldsymbol{w^t}) \\ &+ \int_0^1 (\nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{z}, \boldsymbol{a_k}) - \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k}))^T (\boldsymbol{w} - \boldsymbol{w^t}) d\tau \Big\}_{k=1,2}. \end{aligned} \tag{32}$$

Then, according to

$$\begin{aligned} &\Big| \int_0^1 (\nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{z}, \boldsymbol{a_k}) - \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k}))^T (\boldsymbol{w} - \boldsymbol{w^t}) d\tau \Big| \\ &\leq \int_0^1 \Big| (\nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{z}, \boldsymbol{a_k}) - \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k}))^T (\boldsymbol{w} - \boldsymbol{w^t}) \Big| d\tau, \end{aligned} \tag{33}$$

and by Cauchy-Schwarz inequality, we obtain

$$\begin{aligned} &\left\{ \Big| (\nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{z}, \boldsymbol{a_k}) - \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k}))^T (\boldsymbol{w} - \boldsymbol{w^t}) \Big| \right. \\ &\left. \leq \Big\| \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{z}, \boldsymbol{a_k}) - \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k}) \Big\|_{\boldsymbol{2}} \cdot \Big\| \boldsymbol{w} - \boldsymbol{w^t} \Big\|_{\boldsymbol{2}} \right\}_{k=1,2}. \end{aligned} \tag{34}$$

4.3.2 Linearized Hedging Model

The Hessian captures the second-order behavior of the function, essentially measuring the rate of change of the gradient. If the Hessian is bounded, it implies that the gradient does not change too rapidly, which is crucial for our approximation. The rate of change in the gradients can be quantified by the Hessian of the model function at the current point $\boldsymbol{w^t}$, denoted by $\left\{ \nabla^{\boldsymbol{2}} \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k}) \right\}_{k=1,2}$. Since the Hessian represents

the changing rate of the gradient with respect to the parameter variable $\boldsymbol{w}$, the approximated change induced in the gradient can be expressed as:

$$\left\{\left\|\nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k}) - \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{z}, \boldsymbol{a_k})\right\| \approx \left\|\boldsymbol{z} - \boldsymbol{w^t}\right\| \cdot \left\|\boldsymbol{\nabla^2 f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k})\right\|\right\}_{k=1,2}. \tag{35}$$

If $\nabla^2 f_k(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{a}_k)$ is bounded, we can ensure that the relative change in the gradient is small. Specifically, if the Hessian remains constant along the trajectory, the relative change in the gradient due to this perturbation is

$$\frac{\left\|\nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k}) - \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{z}, \boldsymbol{a_k})\right\|}{\left\|\nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k})\right\|} \approx \frac{\left\|\boldsymbol{z} - \boldsymbol{w^t}\right\| \cdot \left\|\boldsymbol{\nabla^2 f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k})\right\|}{\left\|\nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k})\right\|}. \tag{36}$$

This shows that the relative change in the gradient is proportional to the magnitude of the perturbation $\left\|(\boldsymbol{w} - \boldsymbol{w^t})\right\|$ and inversely proportional to the magnitude of the gradient $\|\nabla f_k(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{a}_k)\|$. This ratio quantifies how much the gradient changes when moving from $\boldsymbol{w^t}$ to another point $\boldsymbol{z}$. If this ratio is small, it indicates that the gradients at $\boldsymbol{w^t}$ and $\boldsymbol{z}$ are similar in magnitude and direction. As long as the Hessian remains bounded, the relative change in the gradient will be sufficiently small, ensuring that the linear approximation of the model function is accurate.

According to the analysis in Section 5, we define β_k as

$$\beta_k = \max\left(A(M_2 H^2 + M_1)^L, A M_2^L\right), \tag{37}$$

and we have

$$\left\{\left\|\boldsymbol{\nabla^2 f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k})\right\| \leq \beta_k\right\}_{k=1,2}. \tag{38}$$

Combining with (27) and (35), we have

$$\left\{\left\|\nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{z}, \boldsymbol{a_k}) - \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k})\right\|_{\boldsymbol{2}} \leq \tau\beta_k \left\|(\boldsymbol{w} - \boldsymbol{w^t})\right\|_{\boldsymbol{2}}\right\}_{k=1,2}, \tag{39}$$

According to (39), we have

$$\left\{\left|\int_0^1 (\nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{z}, \boldsymbol{a_k}) - \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k}))^T (\boldsymbol{w} - \boldsymbol{w^t}) d\tau\right| \leq \frac{\beta_k}{2} \left\|(\boldsymbol{w} - \boldsymbol{w^t})\right\|_{\boldsymbol{2}}^{\boldsymbol{2}}\right\}_{k=1,2}. \tag{40}$$

The relative change in the gradient is then

$$\frac{\left\|\nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k}) - \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{z}, \boldsymbol{a_k})\right\|}{\left\|\nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k})\right\|} \leq \frac{\beta_k \left\|\boldsymbol{z} - \boldsymbol{w^t}\right\|}{\left\|\nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k})\right\|}. \tag{41}$$

When the relative change in the gradient is small, the linear approximation $\hat{f}_k$ accurately represents the behavior of f_k around $\boldsymbol{w^t}$.

As long as the Hessian remains bounded, the relative change in the gradient is small for small perturbations $(\boldsymbol{w} - \boldsymbol{w^t})$. This ensures that the linear approximation defined as

$$\hat{f}_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) = f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w^t}, \boldsymbol{a}_k) + \triangledown f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w^t}, \boldsymbol{a}_k)^T (\boldsymbol{w} - \boldsymbol{w^t}) \tag{42}$$

is accurate. According to the above analysis, the model function can be rewritten as follows

$$\boldsymbol{f_k}(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a_k}) = \boldsymbol{f_k}(\boldsymbol{x}_{t_i}, \boldsymbol{w^t}, \boldsymbol{a_k}) + \nabla \boldsymbol{f_k}(\boldsymbol{x_{t_i}}, \boldsymbol{w^t}, \boldsymbol{a_k})^{\boldsymbol{T}} (\boldsymbol{w} - \boldsymbol{w^t}) + \mathcal{O}\Big(\frac{\beta_k}{2} \left\|(\boldsymbol{w} - \boldsymbol{w^t})\right\|_{\boldsymbol{2}}^{\boldsymbol{2}}\Big). \tag{43}$$

If the relative change is bounded and small for small $\boldsymbol{w} - \boldsymbol{w^t}$, the higher-order terms can be considered insignificant, ensuring that that the linear model $\hat{f}_k$ closely approximates f_k. Combining the bounds, the accuracy can be quantified as follows

$$\left| f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) - \hat{f}_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) \right| \leq \frac{\beta_k}{2} \left\| \boldsymbol{w} - \boldsymbol{w^t} \right\|^2 \tag{44}$$

This shows that the error in the linear approximation is quadratic in the perturbation size $\left\| \boldsymbol{w} - \boldsymbol{w^t} \right\|$ and proportional to the bound β_k on the Hessian. Since this bound is proportional to $\left\| (\boldsymbol{w} - \boldsymbol{w^t}) \right\|^2$, the error introduced by the second-order term becomes negligible for small perturbations. Therefore, the model function f_k is very close to its linear approximation when the Hessian is bounded.

The optimal hedging involves refraining from making trades when the hedge position falls within a designated no-trade interval. When the hedge ratio reaches either endpoint of this interval, the most effective strategy is to execute trades to either maintain the hedge ratio on the boundary or within the interval.

4.4 Anchor Hedge Strategy Using Black-Scholes Delta

The anchor hedge strategy involves using the Black-Scholes delta, which is a derivative of the Black-Scholes formula commonly used in financial mathematics to hedge options. This strategy leverages the sensitivity of the option's price to small changes in the price of the underlying asset, known as the delta, to create a hedged position. Here's a detailed explanation of the components and the process:

The variable bs_{t_i} is defined as

$$bs_{t_i} = \frac{\log(\frac{P_{t_i}}{P_s}) + \frac{\sigma^2 (T - t_i)}{2}}{\sigma \sqrt{T - t_i}} \tag{45}$$

where P_{t_i} is the current price of the underlying asset at time t_i, P_s is the strike price of the option, σ is the volatility of the underlying asset's returns, T is the time to maturity of the option, and t_i is the current time. This equation calculates the standardized distance between the logarithm of the current price and the strike price, adjusted for the time remaining until maturity and the volatility. It essentially measures how far the current price is from the strike price in standard deviation units, taking into account the time decay.

The Black-Scholes delta $\delta_{t_i}^{bs}$ is given by

$$\delta_{t_i}^{bs} = \frac{1}{2} \left(1 + \frac{2}{\sqrt{\pi}} \int_0^{bs_{t_i}} e^{-t^2} dt \right) \tag{46}$$

This represents the hedge ratio, or the sensitivity of the option's price to small changes in the price of the underlying asset. The delta is the probability that the option will end up in-the-money, which means it will have a positive payoff at expiration. The term $\frac{2}{\sqrt{\pi}} \int_0^{bs_{t_i}} e^{-t^2} dt$ is the error function $\text{erf}(bs_{t_i})$, related to the normal distribution and helps in computing the probabilities associated with the normal distribution.

The delta $\delta_{t_i}^{bs}$ ranges from 0 to 1. When the underlying asset price P_{t_i} is much higher than the strike price P_s, bs_{t_i} will be large, and $\delta_{t_i}^{bs}$ will approach 1, indicating a high probability that the option will end up in-the-money. Conversely, if P_{t_i} is much lower than P_s, bs_{t_i} will be negative and large in magnitude, and $\delta_{t_i}^{bs}$ will approach 0, indicating a low probability of the option being in-the-money.

The anchor hedge strategy involves using this delta to hedge an option position. By holding $\delta_{t_i}^{bs}$ units of the underlying asset for each option, the trader can create a hedged position that mitigates the risk from small movements in the underlying asset's price. The delta is recalculated periodically as P_{t_i} and t_i change, to maintain the hedge.

4.5 Model Function and Position Adjustment Strategy

The model function outputs define the boundaries for the upper and lower gaps derived from the anchor hedge strategy. Specifically, we define:

$$\begin{aligned} \Delta_{t_i}^l &= \delta_{t_i}^{bs} - \hat{f}_1(\boldsymbol{x_{t_i}}, \boldsymbol{w}, \boldsymbol{a_1}) \\ \Delta_{t_i}^u &= \delta_{t_i}^{bs} + \hat{f}_2(\boldsymbol{x_{t_i}}, \boldsymbol{w}, \boldsymbol{a_2}). \end{aligned} \tag{47}$$

where $\delta_{t_i}^{bs}$ is the Black-Scholes delta, $\hat{f}_1$ and $\hat{f}_2$ are the linearized model functions representing the lower and upper adjustments to the delta, respectively. The subsequent position, $\delta_{t_{i+1}}$, is determined by:

$$\delta_{t_{i+1}} = \min\{\max\{\Delta_{t_i}^l, \delta_{t_i}\}, \Delta_{t_i}^u\}. \tag{48}$$

This equation ensures that the new position is constrained within the defined upper and lower bounds, providing a controlled and adaptive hedge adjustment.

In this conceptual framework, the introduced feature addresses the challenge of position-dependence by ensuring that the inputs to the neural network remain independent of the current position. This enhancement significantly improves the training process. The strategy encoded by the neural network leverages a transaction band, which is cost-effective as it restricts transactions within the specified band, thereby minimizing unnecessary trading costs. The theoretical foundation supporting this design is robust, demonstrating that the strategy outlined in equations (47)-(48) is optimal not only for European options but also for a broader various utilities and derivatives, including exotic options Imaki et al. (2021).

5 Analysis of Approximation Quality in Hedging Model

To simplify the analysis, we assume each variable in $\{\boldsymbol{a}_k\}_{k=1,2}$ is sampled uniformly at random and exclude it from the training procedure, focusing solely on the weight matrices $\boldsymbol{W}^{(l)}$ and bias vectors $\boldsymbol{b}^{(l)}$, which make up the variable $\boldsymbol{w} = \{\boldsymbol{W}^{(l)}, \boldsymbol{b}^{(l)}\}_{l=1}^{L-1}$. We consider the following assumptions, i.e., the derivative of the activation function $|\sigma'(z)|$ is bounded by M_1 for all z, the input vectors $\boldsymbol{h}^{(l-1)}$ are bounded such that $\|\boldsymbol{h}^{(l-1)}\| \leq H$, and the elements $\boldsymbol{a}_{k,j}$ are constants with a maximum absolute value $A = \max_j |\boldsymbol{a}_{k,j}|$.

5.1 Gradient Magnitude Analysis

To compute the gradient of the model function with respect to the weight matrices $\boldsymbol{W}^{(l)}$ and bias vectors $\boldsymbol{b}^{(l)}$, we use the chain rule, where the terms $\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}^{(i)}\boldsymbol{h}^{(i-1)} + \boldsymbol{b}^{(i)})$ capture the propagation of the gradient through all previous layers. The gradient with respect to the weight matrices $\boldsymbol{W}^{(l)}$ is derived as

$$\nabla_{\boldsymbol{W}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) = \nabla_{\boldsymbol{h}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) \cdot \nabla_{\boldsymbol{W}^{(l)}} \boldsymbol{h}^{(l)}, \tag{49}$$

where

$$\nabla_{\boldsymbol{W}^{(l)}} \boldsymbol{h}^{(l)} = \sigma'(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \cdot \boldsymbol{h}^{(l-1)} \tag{50}$$

Combining (49) and (50), we have

$$\nabla_{\boldsymbol{W}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) = \frac{1}{\sqrt{m}} \sum_{j=1}^{m} \boldsymbol{a}_{k,j} \left(\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}^{(i)}\boldsymbol{h}^{(i-1)} + \boldsymbol{b}^{(i)}) \right) \cdot \sigma'(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \cdot \boldsymbol{h}^{(l-1)}. \tag{51}$$

The gradient with respect to the bias vectors $\boldsymbol{b}^{(l)}$ is derived as

$$\nabla_{\boldsymbol{b}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) = \nabla_{\boldsymbol{h}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) \cdot \nabla_{\boldsymbol{b}^{(l)}} \boldsymbol{h}^{(l)}, \tag{52}$$

where

$$\nabla_{\boldsymbol{b}^{(l)}} \boldsymbol{h}^{(l)} = \sigma'(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}). \tag{53}$$

Combining (52) and (53), we have

$$\nabla_{\boldsymbol{b}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) = \frac{1}{\sqrt{m}} \sum_{j=1}^{m} \boldsymbol{a}_{k,j} \left(\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}^{(i)} \boldsymbol{h}^{(i-1)} + \boldsymbol{b}^{(i)}) \right) \cdot \sigma'(\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}). \tag{54}$$

The product term $\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}^{(i)} \boldsymbol{h}^{(i-1)} + \boldsymbol{b}^{(i)})$ can be bounded by M_1^{l-1}. Next, consider $\sigma'(\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \cdot \boldsymbol{h}^{(l-1)}$. Using the bounds $|\sigma'(\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)})| \leq M_1$ and $\|\boldsymbol{h}^{(l-1)}\| \leq H$, we get

$$|\sigma'(\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \cdot \boldsymbol{h}^{(l-1)}| \leq M_1 H. \tag{55}$$

Then, considering the sum over m terms, we have

$$\left\| \frac{1}{\sqrt{m}} \sum_{j=1}^{m} \boldsymbol{a}_{k,j} \left(\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}^{(i)} \boldsymbol{h}^{(i-1)} + \boldsymbol{b}^{(i)}) \right) \cdot \sigma'(\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \cdot \boldsymbol{h}^{(l-1)} \right\| \leq \frac{1}{\sqrt{m}} \sum_{j=1}^{m} |\boldsymbol{a}_{k,j}| M_1^{l-1} M_1 H. \tag{56}$$

Since $\boldsymbol{a}_{k,j}$ are constants and A is the maximum of their absolute values

$$\frac{1}{\sqrt{m}} \sum_{j=1}^{m} |\boldsymbol{a}_{k,j}| \leq A\sqrt{m}. \tag{57}$$

Combining these bounds, we get

$$\|\nabla_{\boldsymbol{W}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k)\| \leq \frac{1}{\sqrt{m}} \cdot A\sqrt{m} \cdot M_1^l \cdot H = A M_1^l H. \tag{58}$$

Similarly, for the bias vectors, given the gradient expression

$$\nabla_{\boldsymbol{b}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) = \frac{1}{\sqrt{m}} \sum_{j=1}^{m} \boldsymbol{a}_{k,j} \left(\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}^{(i)} \boldsymbol{h}^{(i-1)} + \boldsymbol{b}^{(i)}) \right) \cdot \sigma'(\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}). \tag{59}$$

we can follow a similar analysis. We have:

$$\left\| \frac{1}{\sqrt{m}} \sum_{j=1}^{m} \boldsymbol{a}_{k,j} \left(\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}^{(i)} \boldsymbol{h}^{(i-1)} + \boldsymbol{b}^{(i)}) \right) \cdot \sigma'(\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \right\| \leq \frac{1}{\sqrt{m}} \sum_{j=1}^{m} |\boldsymbol{a}_{k,j}| M_1^l. \tag{60}$$

Using the same argument for $|\boldsymbol{a}_{k,j}|$ in (57), we get

$$\|\nabla_{\boldsymbol{b}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k)\| \leq \frac{1}{\sqrt{m}} \cdot A\sqrt{m} \cdot M_1^l = A M_1^l. \tag{61}$$

The bound of the gradient for both the weight matrices and the bias vectors can be summarized as

$$\|\nabla_{\boldsymbol{W}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k)\| \leq A M_1^l H, \tag{62}$$

and

$$\|\nabla_{\boldsymbol{b}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k)\| \leq A M_1^l. \tag{63}$$

Combining the effects from both the weights and biases, we can write the overall bound on the gradient with respect to the model parameters $\boldsymbol{w}$ as

$$\|\nabla_{\boldsymbol{w}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k)\| \leq A M_1^l (H + 1). \tag{64}$$

The analysis above offers essential insights into the gradient behavior within the hedging model's training process. By applying the chain rule, it is evident that the gradient magnitudes $\|\nabla_{\boldsymbol{w}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k)\|$ are exponentially influenced by the network's depth l. This highlights the importance of selecting appropriate initializations and activation functions. When the magnitude of the activation function's gradient M_1 is controlled around 1, the gradient magnitudes $\|\nabla_{\boldsymbol{w}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k)\|$ tend to stabilize, converging to a constant value as the network depth increases. This behavior strongly supports the subsequent analysis, demonstrating that our approximation maintains high quality, closely aligning with the linearized hedging model.

5.2 Gradient Changing Rate Analysis

To analyze the gradient changing rate, we focus on the rate at which the gradient of the loss function changes with respect to the model parameters. This involves examining the difference in gradients between two different iterations or parameter settings. We start with the gradient of the model function $f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k)$ with respect to the weight matrices $\boldsymbol{W}^{(l)}$ and bias vectors $\boldsymbol{b}^{(l)}$. To measure the gradient changing rate, we consider the difference between the gradients at two different parameter settings, $\boldsymbol{w}$ and $\boldsymbol{w}'$.

For the weight matrices, the gradient difference is

$$\Delta\nabla_{\boldsymbol{W}^{(l)}} f_k = \nabla_{\boldsymbol{W}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) - \nabla_{\boldsymbol{W}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}', \boldsymbol{a}_k). \tag{65}$$

Substituting the gradient expressions, we get

$$\begin{aligned}\Delta\nabla_{\boldsymbol{W}^{(l)}} f_k = \frac{1}{\sqrt{m}} \sum_{j=1}^{m} \boldsymbol{a}_{k,j} \Bigg(&\left(\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}^{(i)}\boldsymbol{h}^{(i-1)} + \boldsymbol{b}^{(i)})\right) \cdot \sigma'(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \cdot \boldsymbol{h}^{(l-1)} \\ &- \left(\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}'^{(i)}\boldsymbol{h}'^{(i-1)} + \boldsymbol{b}'^{(i)})\right) \cdot \sigma'(\boldsymbol{W}'^{(l)}\boldsymbol{h}'^{(l-1)} + \boldsymbol{b}'^{(l)}) \cdot \boldsymbol{h}'^{(l-1)} \Bigg).\end{aligned} \tag{66}$$

Using the triangle inequality, we can bound the difference as

$$\begin{aligned}\|\Delta\nabla_{\boldsymbol{W}^{(l)}} f_k\| \leq \frac{1}{\sqrt{m}} \sum_{j=1}^{m} |\boldsymbol{a}_{k,j}| \Bigg\| &\left(\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}^{(i)}\boldsymbol{h}^{(i-1)} + \boldsymbol{b}^{(i)})\right) \cdot \sigma'(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \cdot \boldsymbol{h}^{(l-1)} \\ &- \left(\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}'^{(i)}\boldsymbol{h}'^{(i-1)} + \boldsymbol{b}'^{(i)})\right) \cdot \sigma'(\boldsymbol{W}'^{(l)}\boldsymbol{h}'^{(l-1)} + \boldsymbol{b}'^{(l)}) \cdot \boldsymbol{h}'^{(l-1)} \Bigg\|.\end{aligned} \tag{67}$$

Similarly, for the bias vectors, the gradient difference is

$$\Delta\nabla_{\boldsymbol{b}^{(l)}} f_k = \nabla_{\boldsymbol{b}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}, \boldsymbol{a}_k) - \nabla_{\boldsymbol{b}^{(l)}} f_k(\boldsymbol{x}_{t_i}, \boldsymbol{w}', \boldsymbol{a}_k). \tag{68}$$

Substituting the gradient expressions, we get

$$\begin{aligned}\Delta\nabla_{\boldsymbol{b}^{(l)}} f_k = \frac{1}{\sqrt{m}} \sum_{j=1}^{m} \boldsymbol{a}_{k,j} \Bigg(&\left(\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}^{(i)}\boldsymbol{h}^{(i-1)} + \boldsymbol{b}^{(i)})\right) \cdot \sigma'(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \\ &- \left(\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}'^{(i)}\boldsymbol{h}'^{(i-1)} + \boldsymbol{b}'^{(i)})\right) \cdot \sigma'(\boldsymbol{W}'^{(l)}\boldsymbol{h}'^{(l-1)} + \boldsymbol{b}'^{(l)}) \Bigg).\end{aligned} \tag{69}$$

Using the triangle inequality, we can bound the difference as

$$\begin{aligned}\|\Delta\nabla_{\boldsymbol{b}^{(l)}} f_k\| \leq \frac{1}{\sqrt{m}} \sum_{j=1}^{m} |\boldsymbol{a}_{k,j}| \Bigg\| &\left(\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}^{(i)}\boldsymbol{h}^{(i-1)} + \boldsymbol{b}^{(i)})\right) \cdot \sigma'(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \\ &- \left(\prod_{i=1}^{l-1} \sigma(\boldsymbol{W}'^{(i)}\boldsymbol{h}'^{(i-1)} + \boldsymbol{b}'^{(i)})\right) \cdot \sigma'(\boldsymbol{W}'^{(l)}\boldsymbol{h}'^{(l-1)} + \boldsymbol{b}'^{(l)}) \Bigg\|.\end{aligned} \tag{70}$$

Let $\Delta\boldsymbol{W}^{(l)} = \boldsymbol{W}^{(l)} - \boldsymbol{W}'^{(l)}$ and $\Delta\boldsymbol{b}^{(l)} = \boldsymbol{b}^{(l)} - \boldsymbol{b}'^{(l)}$. Given the bounds $|\sigma'(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)})| \leq M_1$, $\|\boldsymbol{h}^{(l-1)}\| \leq H$, and $|\boldsymbol{a}_{k,j}| \leq A$, we can derive

$$\|\Delta\nabla_{\boldsymbol{W}^{(l)}} f_k\| \leq \frac{1}{\sqrt{m}} \sum_{j=1}^{m} AM_1^{l-1} M_1 H \|\Delta\boldsymbol{W}^{(l)}\| = AM_1^l H \|\Delta\boldsymbol{W}^{(l)}\|, \tag{71}$$

and

$$\|\Delta\nabla_{\boldsymbol{b}^{(l)}} f_k\| \leq \frac{1}{\sqrt{m}} \sum_{j=1}^{m} A M_1^{l-1} M_1 \|\Delta \boldsymbol{b}^{(l)}\| = A M_1^l \|\Delta \boldsymbol{b}^{(l)}\|. \tag{72}$$

To include the Hessian matrix in our analysis, we use the Taylor expansion around the parameter setting $\boldsymbol{w}$

$$\nabla_{\boldsymbol{W}^{(l)}} f_k(\boldsymbol{w}') \approx \nabla_{\boldsymbol{W}^{(l)}} f_k(\boldsymbol{w}) + \nabla^2_{\boldsymbol{W}^{(l)}} f_k(\boldsymbol{w}) \cdot (\boldsymbol{W}'^{(l)} - \boldsymbol{W}^{(l)}), \tag{73}$$

where $\nabla^2_{\boldsymbol{W}^{(l)}} f_k(\boldsymbol{w})$ denotes the Hessian matrix of the loss function with respect to $\boldsymbol{W}^{(l)}$ at $\boldsymbol{w}$. Therefore, the gradient difference is

$$\Delta\nabla_{\boldsymbol{W}^{(l)}} f_k = \nabla^2_{\boldsymbol{W}^{(l)}} f_k(\boldsymbol{w}) \cdot (\boldsymbol{W}'^{(l)} - \boldsymbol{W}^{(l)}). \tag{74}$$

The norm of this gradient difference is

$$\|\Delta\nabla_{\boldsymbol{W}^{(l)}} f_k\| = \left\| \nabla^2_{\boldsymbol{W}^{(l)}} f_k(\boldsymbol{w}) \cdot (\boldsymbol{W}'^{(l)} - \boldsymbol{W}^{(l)}) \right\|. \tag{75}$$

Similarly, for the bias vectors, we have

$$\Delta\nabla_{\boldsymbol{b}^{(l)}} f_k = \nabla^2_{\boldsymbol{b}^{(l)}} f_k(\boldsymbol{w}) \cdot (\boldsymbol{b}'^{(l)} - \boldsymbol{b}^{(l)}). \tag{76}$$

The norm of the gradient difference for the bias vectors is

$$\|\Delta\nabla_{\boldsymbol{b}^{(l)}} f_k\| = \left\| \nabla^2_{\boldsymbol{b}^{(l)}} f_k(\boldsymbol{w}) \cdot (\boldsymbol{b}'^{(l)} - \boldsymbol{b}^{(l)}) \right\|. \tag{77}$$

Let $\Delta\boldsymbol{W}^{(l)} = \boldsymbol{W}^{(l)} - \boldsymbol{W}'^{(l)}$ and $\Delta\boldsymbol{b}^{(l)} = \boldsymbol{b}^{(l)} - \boldsymbol{b}'^{(l)}$, and given the bounds on the Hessian, we can derive

$$\|\Delta\nabla_{\boldsymbol{W}^{(l)}} f_k\| \leq \|\nabla^2_{\boldsymbol{W}^{(l)}} f_k(\boldsymbol{w})\| \cdot \|\Delta\boldsymbol{W}^{(l)}\|, \tag{78}$$

and

$$\|\Delta\nabla_{\boldsymbol{b}^{(l)}} f_k\| \leq \|\nabla^2_{\boldsymbol{b}^{(l)}} f_k(\boldsymbol{w})\| \cdot \|\Delta\boldsymbol{b}^{(l)}\|. \tag{79}$$

5.3 Hessian Bound Analysis

To bound the Hessian of the model function $f_k(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{a}_k)$ with respect to the weight matrices $\boldsymbol{W}^{(l)}$ and bias vectors $\boldsymbol{b}^{(l)}$, we need to consider the maximum values that the second-order partial derivatives can attain.

First, for the weight matrices $\boldsymbol{W}^{(l)}$, the second-order partial derivatives are given by

$$\nabla^2_{\boldsymbol{W}^{(l)}} f_k(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{a}_k) = \frac{1}{\sqrt{m}} \sum_{j=1}^{m} \boldsymbol{a}_{k,j} \prod_{l=1}^{L} \left[\sigma''(\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \boldsymbol{h}^{(l-1)} (\boldsymbol{h}^{(l-1)})^T + \sigma'(\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \right]. \tag{80}$$

For the bias vectors $\boldsymbol{b}^{(l)}$, the second-order partial derivatives are

$$\nabla^2_{\boldsymbol{b}^{(l)}} f_k(\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{a}_k) = \frac{1}{\sqrt{m}} \sum_{j=1}^{m} \boldsymbol{a}_{k,j} \prod_{l=1}^{L} \left[\sigma''(\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \right]. \tag{81}$$

By making another assumption that the activation function σ and its derivatives σ'' are bounded, where $|\sigma''(z)| \leq M_2$ for all z. The norm of each term in the product can be bounded by

$$\left| \sigma''(\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \boldsymbol{h}^{(l-1)} (\boldsymbol{h}^{(l-1)})^T \right| \leq M_2 H^2 \tag{82}$$

and

$$\left| \sigma'(\boldsymbol{W}^{(l)} \boldsymbol{h}^{(l-1)} + \boldsymbol{b}^{(l)}) \right| \leq M_1. \tag{83}$$

Thus, the norm of the product term is

$$\left|\prod_{l=1}^{L}\left[\sigma''(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)}+\boldsymbol{b}^{(l)})\boldsymbol{h}^{(l-1)}(\boldsymbol{h}^{(l-1)})^T+\sigma'(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)}+\boldsymbol{b}^{(l)})\right]\right| \leq (M_2H^2+M_1)^L. \tag{84}$$

Since $\boldsymbol{a}_{k,j}$ are constants, the sum is bounded by

$$\begin{aligned}&\left|\frac{1}{\sqrt{m}}\sum_{j=1}^{m}\boldsymbol{a}_{k,j}\prod_{l=1}^{L}\left[\sigma''(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)}+\boldsymbol{b}^{(l)})\boldsymbol{h}^{(l-1)}(\boldsymbol{h}^{(l-1)})^T+\sigma'(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)}+\boldsymbol{b}^{(l)})\right]\right| \\ &\leq \frac{1}{\sqrt{m}}\sum_{j=1}^{m}|\boldsymbol{a}_{k,j}|(M_2H^2+M_1)^L.\end{aligned} \tag{85}$$

Letting $A=\max_j|\boldsymbol{a}_{k,j}|$, we get

$$\left|\nabla^2_{\boldsymbol{W}^{(l)}}f_k\right| \leq \frac{1}{\sqrt{m}}mA(M_2H^2+M_1)^L = A(M_2H^2+M_1)^L. \tag{86}$$

Next, consider the Hessian for the bias vectors. Using the bound on σ'', we have

$$\left|\prod_{l=1}^{L}\sigma''(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)}+\boldsymbol{b}^{(l)})\right| \leq M_2^L. \tag{87}$$

The sum is bounded by

$$\left|\frac{1}{\sqrt{m}}\sum_{j=1}^{m}\boldsymbol{a}_{k,j}\prod_{l=1}^{L}\sigma''(\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l-1)}+\boldsymbol{b}^{(l)})\right| \leq \frac{1}{\sqrt{m}}\sum_{j=1}^{m}|\boldsymbol{a}_{k,j}|M_2^L. \tag{88}$$

Combining with $A=\max_j|\boldsymbol{a}_{k,j}|$, we get

$$\left|\nabla^2_{\boldsymbol{b}^{(l)}}f_k\right| \leq \frac{1}{\sqrt{m}}mAM_2^L = AM_2^L. \tag{89}$$

Moreover, the mixed second-order partial derivatives $\nabla^2_{\boldsymbol{b}^{(l)},\boldsymbol{W}^{(l)}}f_k$ and $\nabla^2_{\boldsymbol{W}^{(l)},\boldsymbol{b}^{(l)}}f_k$ are zero due to the structure of the model function and the independence of the weight matrices $\boldsymbol{W}^{(l)}$ and the bias vectors $\boldsymbol{b}^{(l)}$. The weight matrix $\boldsymbol{W}^{(l)}$ and the bias vector $\boldsymbol{b}^{(l)}$ at the same layer l do not directly interact in the second-order sense. Therefore, their partial derivatives are computed independently, resulting in zero mixed second-order derivatives

$$\frac{\partial}{\partial\boldsymbol{b}^{(l)}}\left(\frac{\partial f_k}{\partial\boldsymbol{W}^{(l)}}\right)=0 \quad \text{and} \quad \frac{\partial}{\partial\boldsymbol{W}^{(l)}}\left(\frac{\partial f_k}{\partial\boldsymbol{b}^{(l)}}\right)=0. \tag{90}$$

The overall Hessian norm is bounded by considering the maximum bounds of the components

$$\left\|\nabla^2 f_k(\boldsymbol{x},\boldsymbol{w},\boldsymbol{a}_k)\right\| \leq \max\left(A(M_2H^2+M_1)^L, AM_2^L\right). \tag{91}$$

6 Numerical Results

The strategies compared include the proposed Deep Hedging with Linearized Neural Networks (DHLNN), Deep Hedging with Neural Tangent Bootstrap (DHNTB), Deep Hedging with Multi-Layer Perceptron (DHMLP), and Black-Scholes Delta Hedging (BSDH). We conducted numerical simulations and experiments to evaluate the effectiveness of our proposed hedging method. For comparison, we utilized the Black-Scholes Delta Hedging (BSDH) strategy, a fundamental technique in options trading that involves continuously adjusting the number of shares in the underlying asset to maintain a delta-neutral portfolio Hull & Basu (2016)Broadie et al. (1999). This approach minimizes risk from price fluctuations in the underlying asset and serves as a standard benchmark in our experiments. We also included state-of-the-art deep hedging baselines, DHMLP and DHNTB, as described in previous studies Buehler et al. (2019)Imaki et al. (2021). Our neural network model was designed with four hidden layers, each containing 64 neurons, to strike a balance between performance and computational efficiency.

6.1 Datasets Used in Experiments

We have tow different sources of data used to evaluate the performance of our designed deep hedging strategy, i.e., simulated price trajectories from Imaki et al. (2021) and price trajectories from real market data from Andrew Meyer (2021). Based on these price trajectories, we applied two kinds of options into the hedging strategies, i.e., European Call Options and Lookback Call Options.

6.1.1 Price Trajectories

We first simulate market data by modeling the spot prices of the underlying assets following a geometric Brownian motion. The use of simulated market data is crucial because real market data often cannot cover the full range of conditions necessary to comprehensively test a hedging strategy. By simulating market conditions, we can evaluate the robustness of our hedging strategy across all possible scenarios it may encounter in reality, ensuring that it performs well even under extreme market behaviors.

To rigorously assess the strategy, we manipulate key parameters such as the volatility and drift of the spot price, and the transaction cost rate. At each time step, the relevant information considered includes log-moneyness, time to maturity, and volatility. The derivatives are set with a maturity of $T = 30/365$, with time steps defined as $t = 0, 1/365, \ldots, 30/365$. For each iteration, we generate $10,000$ Monte Carlo trajectories to represent the underlying asset prices, which are then used as training data samples Imaki et al. (2021). To ensure a rigorous evaluation, we simulate an additional set of $10,000$ trajectories to serve as testing data. The final assessment of our strategy's effectiveness is based on the terminal hedging performance evaluated using these testing data trajectories.

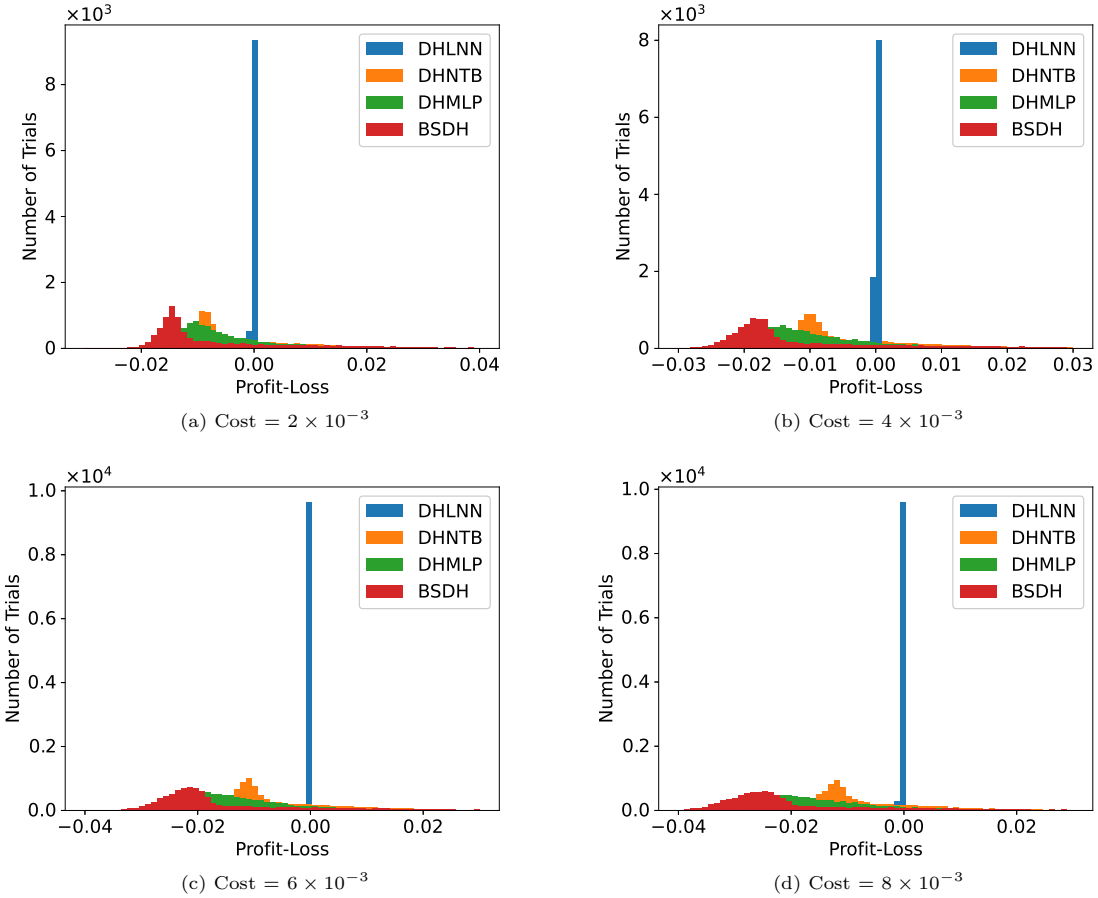
For the real market data set, which comprises highly detailed financial data for numerous stocks across various sectors, containing hundreds of millions of rows. Central to this data is the order book, an electronic ledger of buy and sell orders for specific securities, meticulously organized by price levels. The order book is indispensable for market research as it mirrors market demand and supply, offering a real-time snapshot of trading intentions. From the order book data, we can derive several crucial statistics to assess market liquidity and stock valuation. One of the most significant metrics is the bid-ask spread, which is calculated from the best bid and offer prices. This spread is essential for understanding the immediate costs associated with trading. Another vital metric is the Weighted Average Price (WAP), which plays a critical role in our analysis as it serves as our measure for price trajectories. WAP is computed by considering both the price and volume of top-level bids and offers, providing a comprehensive, instantaneous valuation that reflects the true market conditions. By accounting for market supply and demand, WAP allows us to track price movements more accurately and assess the statistical volatility of the assets over time.

6.1.2 Options in Experiments

In our experiments, we apply two distinct types of options, i.e., the European Call Option and the Lookback Call Option, to evaluate the performance of our hedging strategy. The choice of these two options is strategic, as each provides unique insights into the strategy's effectiveness under different market conditions and complexities.

The first option we use is the European Call Option. This is a standard derivative where the holder has the right, but not the obligation, to purchase the underlying asset at a predetermined strike price on a specified maturity date. The European Call Option is a fundamental choice for testing because of its simplicity and prevalence in financial markets. By using this option, we can evaluate our hedging strategy's ability to manage risk in a straightforward scenario, where the primary concern is the asset's price at the time of maturity. This option helps us understand how well the strategy performs under typical market conditions without the complexity of path-dependent features.

The second option employed is the Lookback Call Option, which is a more complex, path-dependent derivative. Unlike the European Call Option, the payoff of a Lookback Call Option depends on the maximum price of the underlying asset during the option's life, allowing the holder to "look back" and choose the most favorable price. This characteristic makes the Lookback Call Option particularly sensitive to the entire price trajectory of the underlying asset, not just the price at maturity. By testing our hedging strategy with this option,



(a) Cost $= 2 \times 10^{-3}$

(b) Cost $= 4 \times 10^{-3}$

(c) Cost $= 6 \times 10^{-3}$

(d) Cost $= 8 \times 10^{-3}$

Figure 1: Comparison of hedging performance across different transaction costs $\{2 \times 10^{-3}, 4 \times 10^{-3}, 6 \times 10^{-3}, 8 \times 10^{-3}\}$ for a European option with a strike price of 1.2. The analysis, conducted over 50 training epochs, focuses on the distribution of hedging PNL with volatility set at 0.1.

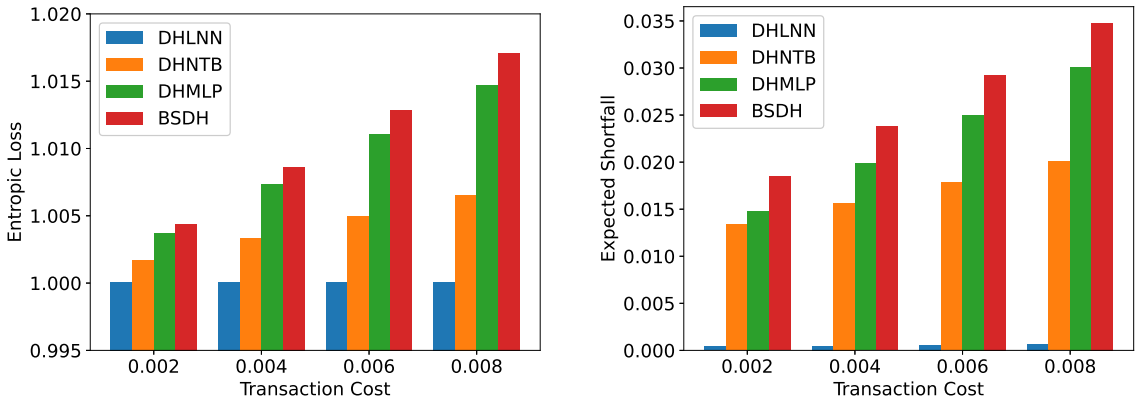
we can assess its robustness in more challenging scenarios, where the price path plays a critical role. This option is ideal for evaluating how well the strategy handles scenarios involving significant price volatility and provides a more comprehensive test of its effectiveness.

6.2 Experiments with Stochastically Simulated Market Data

We first provide compelling evidence of the DHLNN's effectiveness in managing transaction costs while preserving hedging performance. This capability is especially crucial in high-cost environments, where traditional and other deep hedging strategies falter. The analysis highlights the importance of developing adaptive and robust hedging models, capable of maintaining efficiency across varying transaction cost scenarios.

In Fig. 1, the analysis compares the hedging performance across different transaction costs $\{2 \times 10^{-3}, 4 \times 10^{-3}, 6 \times 10^{-3}, 8 \times 10^{-3}\}$ for a European option with a strike price of 1.2. The evaluation is based on the distribution of hedging profit and loss (PNL) after training the deep hedging models for 50 epochs, with volatility set at 0.1.

As transaction costs increase, the hedging PNL distributions exhibit a clear degradation in performance across all models. This trend highlights the sensitivity of hedging strategies to transaction costs, a crucial factor in real-world trading scenarios. The results illustrate that the proposed DHLNN method consistently



(a) (b)

Figure 2: Hedging Performance comparison over different transaction costs $\{2 \times 10^{-3}, 4 \times 10^{-3}, 6 \times 10^{-3}, 8 \times 10^{-3}\}$ of underlying asset for a European option and strike 1.2 with 50 training epochs for the Entropic Loss and Expected Shortfall of Hedging PNL where volatility is set as 0.1.

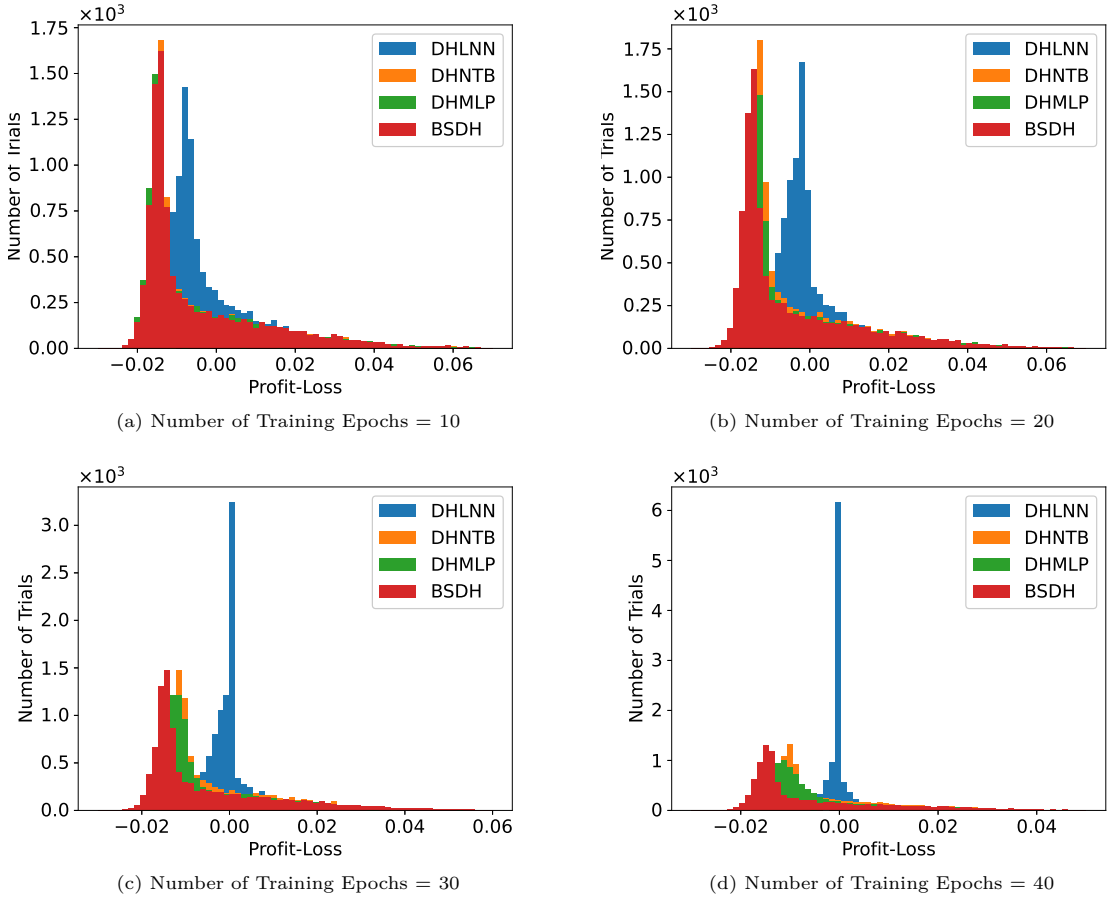
outperforms the baselines, particularly at higher transaction cost levels. This superior performance indicates the effectiveness of DHLNN in minimizing transaction costs while maintaining robust hedging performance.

When transaction costs are low as 2×10^{-3}, the differences between the models are relatively modest, with all strategies showing strong hedging performance. However, as transaction costs rise, the performance gap widens significantly. The baseline models, such as the BSDH, experience a noticeable decline in PNL distribution as transaction costs increase. At the 6×10^{-3} cost level, the number of trials with losses beyond -0.02 becomes prominent, indicating that BSDH is less efficient at higher transaction costs. This decline underscores the limitations of traditional hedging strategies in environments with substantial transaction costs. On the other hand, the DHLNN's ability to adapt to different transaction cost levels demonstrates its robustness and flexibility, particularly when compared to state-of-the-art deep hedging models like DHMLP and DHNTB, which have more spread-out distributions, indicating less efficient handling of transaction costs compared to DHLNN.

Fig. 2 presents a detailed analysis of the hedging performance of various strategies under different transaction costs 2×10^{-3}, 4×10^{-3}, 6×10^{-3}, and 8×10^{-3} for a European option with a strike price of 1.2. The metrics evaluated include Entropic Loss and Expected Shortfall, both of which are crucial for understanding the effectiveness of hedging strategies in managing risk and minimizing potential losses.

In the case of Entropic Loss as shown in Fig. 2(a), which reflects the risk-adjusted return of each strategy, DHLNN consistently outperforms the other strategies across all levels of transaction costs. For instance, at the highest transaction cost 8×10^{-3}), DHLNN records an Entropic Loss just above 1.005, significantly lower than that of the other strategies. This indicates that DHLNN manages risk more effectively, optimizing the trade-off between risk and return. Meanwhile, BSDH exhibits the highest Entropic Loss, particularly at higher transaction costs, reaching approximately 1.015, underscoring its inefficiency due to continuous rebalancing, which incurs greater costs. DHMLP and DHNTB perform moderately well but do not achieve the same efficiency as DHLNN, especially as transaction costs rise.

The Expected Shortfall analysis as shown in Fig. 2(b), which assesses the worst-case scenario risk by evaluating the average loss in the tail of the distribution, further highlights the robustness of DHLNN. Across all transaction costs, DHLNN maintains the lowest Expected Shortfall, with values around 0.018 even at the highest transaction cost level 8×10^{-3}). This indicates DHLNN's superior ability to mitigate extreme losses. BSDH, on the other hand, shows the highest Expected Shortfall, particularly at higher transaction costs, where it reaches around 0.034, demonstrating its vulnerability to extreme losses. DHMLP and DHNTB offer better performance than BSDH but still fall short of the robustness displayed by DHLNN, with increasing Expected Shortfall as transaction costs rise.



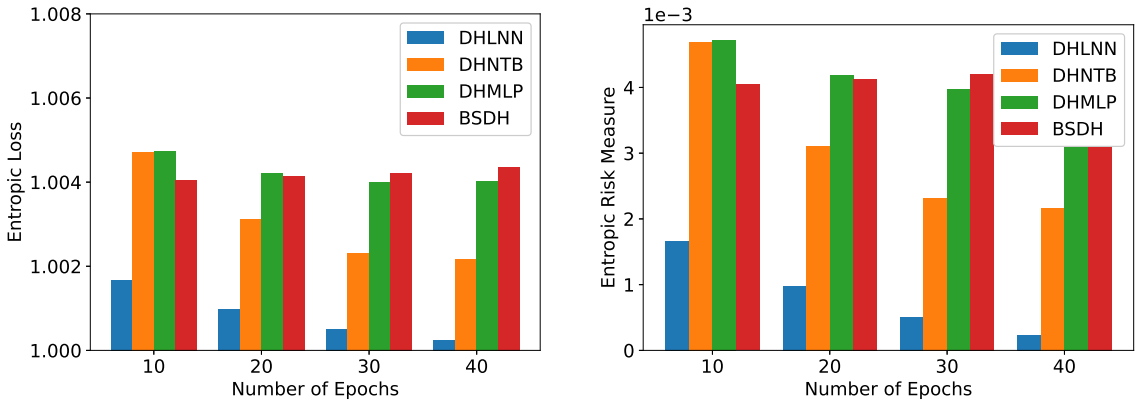
(a) Number of Training Epochs = 10

(b) Number of Training Epochs = 20

(c) Number of Training Epochs = 30

(d) Number of Training Epochs = 40

Figure 3: Comparison of convergence performance for deep hedging models across different training epochs $\{10, 20, 30, 40\}$ on a European option with a strike price of 1.2. The analysis includes transaction costs of 2×10^{-3} and a fixed volatility of 0.1.



(a) (b)

Figure 4: Comparison of convergence performance across different training epochs $\{10, 20, 30, 40\}$ for deep hedging models applied to a European option with a strike price of 1.2. The evaluation focuses on Entropic Loss and Entropic Risk Measure of hedging PNL. The transaction cost is set at 2×10^{-3} and volatility at 0.1.

In Fig. 3, we compare the PNL distributions of various deep hedging models across different training epochs $10, 20, 30, 40$ for European option with a strike price of 1.2, under transaction costs of 2×10^{-3} and a fixed volatility of 0.1. The DHLNN strategy consistently shows a tightly concentrated PNL distribution across all training epochs, demonstrating its strong ability to minimize hedging errors and manage risk effectively. As training progresses, the PNL distribution of DHLNN becomes even more concentrated around zero, indicating stable and reliable performance with increased training, which reflects its robust learning capability.

In contrast, the DHMLP and DHNTB strategies display broader PNL distributions, particularly in the earlier epochs, indicating higher variability and less effective risk management. However, as training continues, these models gradually improve, with their PNL distributions becoming narrower, though they do not reach the same level of concentration as DHLNN. This suggests that while DHMLP and DHNTB benefit from additional training, they require more epochs to achieve a similar performance level. The traditional strategy BSDH shows the least concentrated PNL distributions, with significant tails that suggest a higher likelihood of large deviations, underscoring its limitations in effectively hedging the option. DHLNN outperforms the other models, demonstrating superior risk management and consistency in PNL distribution across different training stages.

We then analyze how different deep hedging models, including DHLNN, DHNTB, and DHMLP, evolve and optimize their performance across increasing training epochs, focusing on Entropic Loss and Entropic Risk Measure as shown in Fig. 4. In the early stages of training, at 10 and 20 epochs, these models are still in the process of adjusting their parameters to fit the market dynamics present in the training data. During this phase, both Entropic Loss and Entropic Risk Measure are relatively high, reflecting the initial challenges these models face in effective hedging and risk management.

As training progresses to around 30 epochs, DHLNN shows substantial improvement, with significant reductions in both Entropic Loss and Entropic Risk Measure. This suggests that DHLNN has effectively optimized its hedging strategy, resulting in more stable and profitable outcomes, with Entropic Loss potentially decreasing by about 20%. By 30 epochs, DHNTB and DHMLP typically reach a plateau, where additional training to 40 epochs yields diminishing returns, indicating that they have reached their best limits for the given market conditions. Meanwhile, our proposed DHLNN continues to improve with much better performance. In contrast, baseline strategies like BSDH, which does not adapt through learning, show stable but generally higher Entropic Loss and Risk Measures, underscoring the advantages of deep learning-based approaches like DHLNN in dynamic market conditions.

We then evaluate the hedging strategies based on the Expected Shortfall of the hedging PNL, as depicted in Fig. 5. The DHLNN model shows marked improvement in minimizing Expected Shortfall as training epochs increase. Initially, the Expected Shortfall is relatively high, reflecting greater tail risk, but it stabilizes at

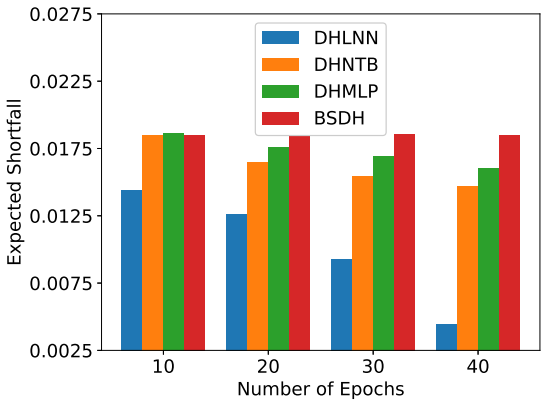


Figure 5: Convergence Performance comparison over different training epochs $\{10, 20, 30, 40\}$ for deep hedging models over a European option with strike 1.2 for Expected Shortfall of hedging PNL, transaction cost is 2×10^{-3} and volatility is set as 0.1.

a lower level after 30 epochs, indicating DHLNN's enhanced ability to manage extreme market scenarios effectively. This rapid convergence suggests DHLNN is highly efficient at learning optimal hedge adjustments to minimize large potential losses.

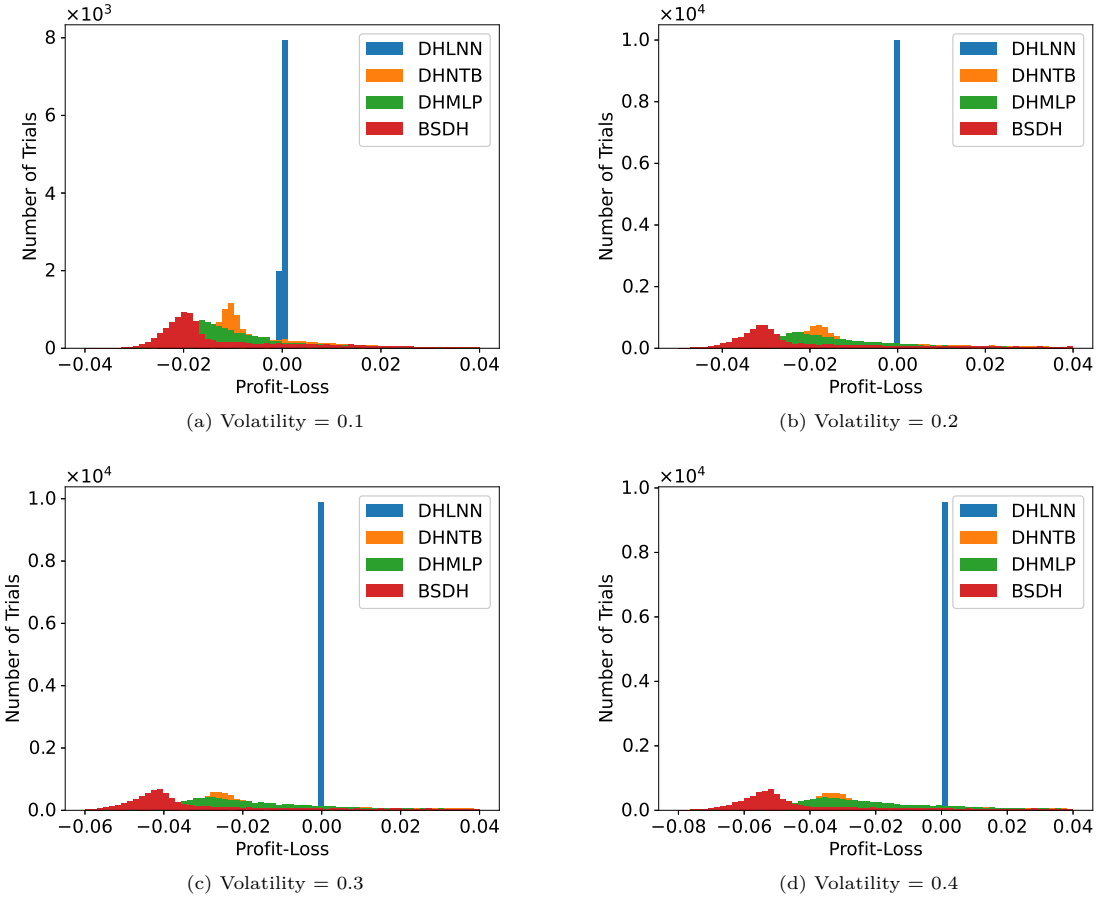
In comparison, DHMLP and DHNTB also show a decrease in Expected Shortfall with more training epochs, though their improvement rate is slower than DHLNN. By around 30 epochs, their Expected Shortfall levels off, indicating diminishing returns with further training. This suggests that while DHMLP and DHNTB can reduce tail risk, they are less efficient than DHLNN. Traditional strategy like BSDH shows consistent Expected Shortfall across all epochs. However, its Expected Shortfall values are generally higher than those of the deep hedging models, particularly DHLNN. The results underscore DHLNN's clear advantage in reducing Expected Shortfall, demonstrating its superior risk management capabilities, especially in mitigating extreme losses, which is crucial for maintaining financial stability in volatile markets.

In Fig. 6, we analyze the PNL distributions of various hedging strategies, including the traditional BSDH, state-of-the-art deep hedging baselines DHMLP and DHNTB, and our proposed DHLNN, under different volatility levels, i.e., 0.1, 0.2, 0.3, and 0.4, in the underlying asset price.

At the lowest volatility of 0.1, all strategies show relatively tight PNL distributions, indicative of a stable market. However, DHLNN already exhibits a great edge by maintaining a more concentrated distribution around the zero PNL mark, suggesting more precise hedging and minimized deviations from expected outcomes. As volatility rises to 0.2, the PNL distributions widen, reflecting increased market uncertainty for all baselines. DHLNN continues to outperform other strategies, with a more focused PNL distribution, indicating better management of moderate market fluctuations and enhanced protection against potential losses.

At a volatility of 0.3, the differences between the strategies become more pronounced. DHLNN's PNL distribution remains relatively narrow, showcasing its robustness in more volatile conditions. In contrast, the BSDH strategy shows significantly wider distributions, indicating reduced hedging effectiveness. While DHMLP and DHNTB perform better than traditional methods, they still do not match DHLNN's superior performance. In the highest volatility scenario of 0.4, all strategies exhibit their widest PNL distributions, but DHLNN leads with a distribution that remains more concentrated than the others. The results highlight DHLNN's resilience in highly volatile markets, where the risk of extreme losses is greatest. DHLNN consistently demonstrates fewer extreme losses and more stable performance across all volatility levels, outperforming both traditional and deep hedging baselines.

Fig. 7 compares the performance of hedging strategies, including traditional BSDH, deep hedging methods DHMLP and DHNTB, and our proposed DHLNN, focusing on Entropic Loss and Entropic Risk Measure across four volatility scenarios as 0.1, 0.2, 0.3, and 0.4.



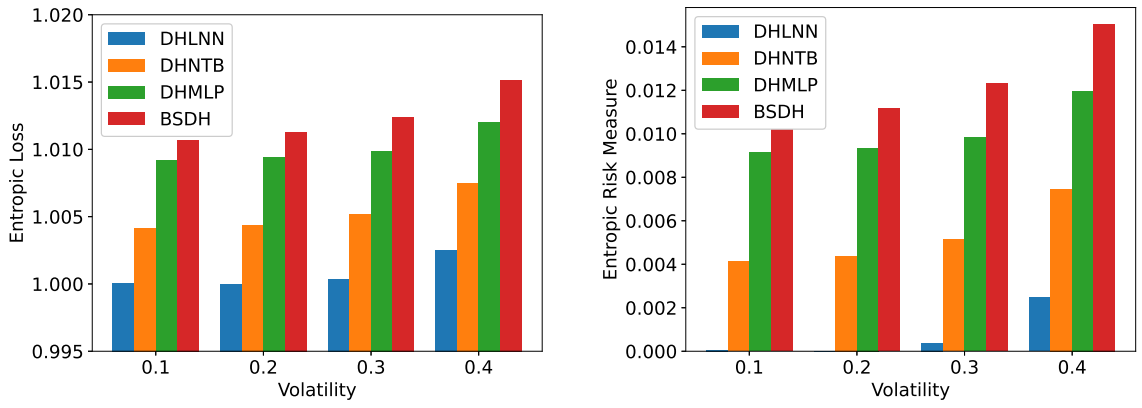
(a) Volatility = 0.1

(b) Volatility = 0.2

(c) Volatility = 0.3

(d) Volatility = 0.4

Figure 6: Hedging Performance comparison of PNL distributions over different volatility of the underline asset price $\{0.1, 0.2, 0.3, 0.4\}$ for a European option and strike 1.2 with transaction cost 5×10^{-3}, with 50 training epochs.



(a)

(b)

Figure 7: Comparison of hedging performance based on PNL distributions across various volatilities of the underlying asset price $\{0.1, 0.2, 0.3, 0.4\}$ for a European option with a strike price of 1.2. The analysis focuses on Entropic Loss and Entropic Risk Measure of the hedging PNL, with a transaction cost of 5×10^{-3} and conducted over 50 training epochs.

As shown in Fig. 7(a), Entropic Loss increases with rising volatility across all strategies, indicating heightened market risk. At a volatility of 0.1, all strategies show minimal Entropic Loss, with DHLNN outperforming the others greatly, suggesting its superior risk management under stable conditions. As volatility rises to 0.2, Entropic Loss increases for all strategies, but DHLNN maintains a lower Entropic Loss than BSDH, DHMLP, and DHNTB, demonstrating its effectiveness in managing escalating risk. In highly volatile environments as 0.3 and 0.4, DHLNN consistently achieves the lowest Entropic Loss, underscoring its robustness in volatile markets. The Entropic Risk Measure, presented in Fig. 7(b), offers a broader view of strategy effectiveness under varying volatility. At a volatility of 0.1 and 0.2, DHLNN shows great advantage over all other strategies exhibiting low Entropic Risk closer to 0. In more volatile scenarios, the Entropic Risk Measure rises sharply, especially for BSDH, while DHLNN consistently exhibits a lower risk measure, indicating better risk-adjusted performance.

6.3 Experiments with Real Market Data

Lookback options, which base their payoff on the maximum price of the underlying asset during the option's life, present significant hedging challenges due to their path dependency. This complexity requires sophisticated hedging strategies capable of adapting to the entire price history of the underlying asset. Traditional strategies like BSDH, which rely on continuous rebalancing based on the current asset price, struggle with Lookback options because they do not account for the historical price extremes that determine the payoff. These limitations highlight the need for more advanced approaches. Our analysis compares deep hedging strategies, specifically DHMLP, DHNTB, and our proposed DHLNN, that are better suited to manage the path dependency and nonlinearities inherent in Lookback options.

In Fig. 8, we evaluate deep hedging strategies for Lookback options, focusing on PNL distributions under varying transaction costs. The strategies compared are DHMLP, DHNTB, and the proposed DHLNN, across transaction costs of 2×10^{-3}, 4×10^{-3}, 6×10^{-3}, and 8×10^{-3}.

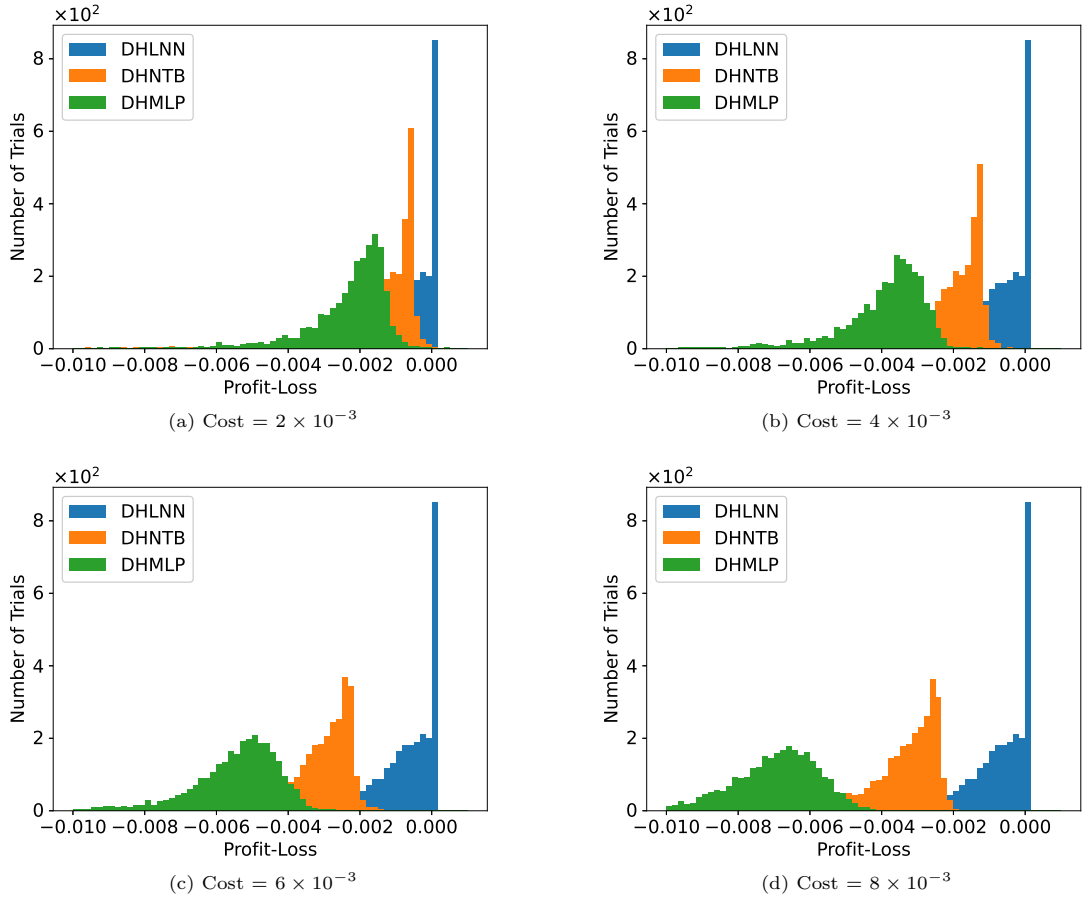
At the lowest cost 2×10^{-3}, all strategies show tight PNL distributions, with DHLNN displaying superior concentration near zero PNL, highlighting its efficiency. As costs increase to 4×10^{-3}, PNL distributions widen, but DHLNN remains more concentrated, indicating resilience. With a further rise to 6×10^{-3}, the impact of transaction costs becomes more pronounced, yet DHLNN continues to outperform in maintaining a tighter distribution. At the highest cost 8×10^{-3}, while all strategies exhibit their widest distributions, DHLNN still demonstrates robust risk management, outperforming the other methods. These results underscore DHLNN's advantages in managing Lookback options, consistently showing tighter PNL distributions and better resilience to rising costs.

Fig. 9 assesses hedging performance based on Entropic Loss and Entropic Risk Measure across the same cost levels. Initially, DHMLP and DHNTB perform well, but as costs rise, their effectiveness declines, with increased Entropic Loss and Risk Measure. DHLNN, however, maintains superior performance, managing the trade-offs between risk and transaction costs effectively. At the highest transaction cost, the baselines struggle significantly, while DHLNN continues to manage risks effectively, demonstrating robustness.

Fig. 10 compares Expected Shortfall across the same cost levels. DHLNN consistently shows minimal Expected Shortfall, even as costs rise, while DHMLP and DHNTB become increasingly vulnerable. This highlights DHLNN's robustness and adaptability in managing tail risk for Lookback options, making it a more reliable strategy under varying cost scenarios.

In Fig. 11, we compare the convergence performance of DHMLP, DHNTB, and our proposed DHLNN for a Lookback option with a strike price of 1.0, across training epochs 10, 20, 30, and 40 under a transaction cost of 2×10^{-3}. We evaluate how each model's hedging performance evolves with increased training with the Lookback options.

Initially, DHMLP performs well, showing a tighter PNL distribution and closer to 0 compared with DHNTB. However, as training extends to 30 and 40 epochs, DHNTB improves significantly, surpassing DHMLP in performance. This suggests that DHNTB benefits more from extended training, refining its hedging strategies over time. Despite these improvements, our proposed DHLNN consistently outperforms both DHMLP and DHNTB across all training settings. DHLNN quickly converges to a tighter PNL distribution, indicating



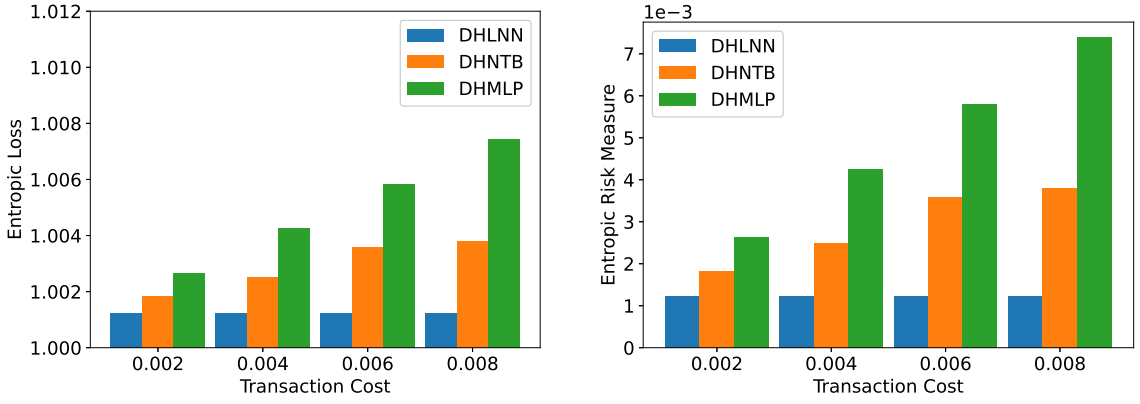
(a) Cost $= 2 \times 10^{-3}$

(b) Cost $= 4 \times 10^{-3}$

(c) Cost $= 6 \times 10^{-3}$

(d) Cost $= 8 \times 10^{-3}$

Figure 8: Hedging Performance comparison over different transaction costs $\{2 \times 10^{-3}, 4 \times 10^{-3}, 6 \times 10^{-3}, 8 \times 10^{-3}\}$ of underlying asset for a Lookback option and strike 1.0 with 50 training epochs for the distribution of Hedging PNL.



(a)

(b)

Figure 9: Comparison of hedging performance across various transaction costs $\{2 \times 10^{-3}, 4 \times 10^{-3}, 6 \times 10^{-3}, 8 \times 10^{-3}\}$ for a Lookback option with a strike price of 1.0. The evaluation uses 50 training epochs and focuses on the Entropic Loss and Entropic Risk Measure of the hedging PNL.

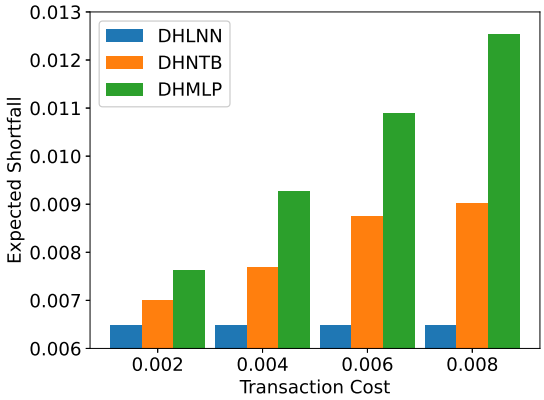
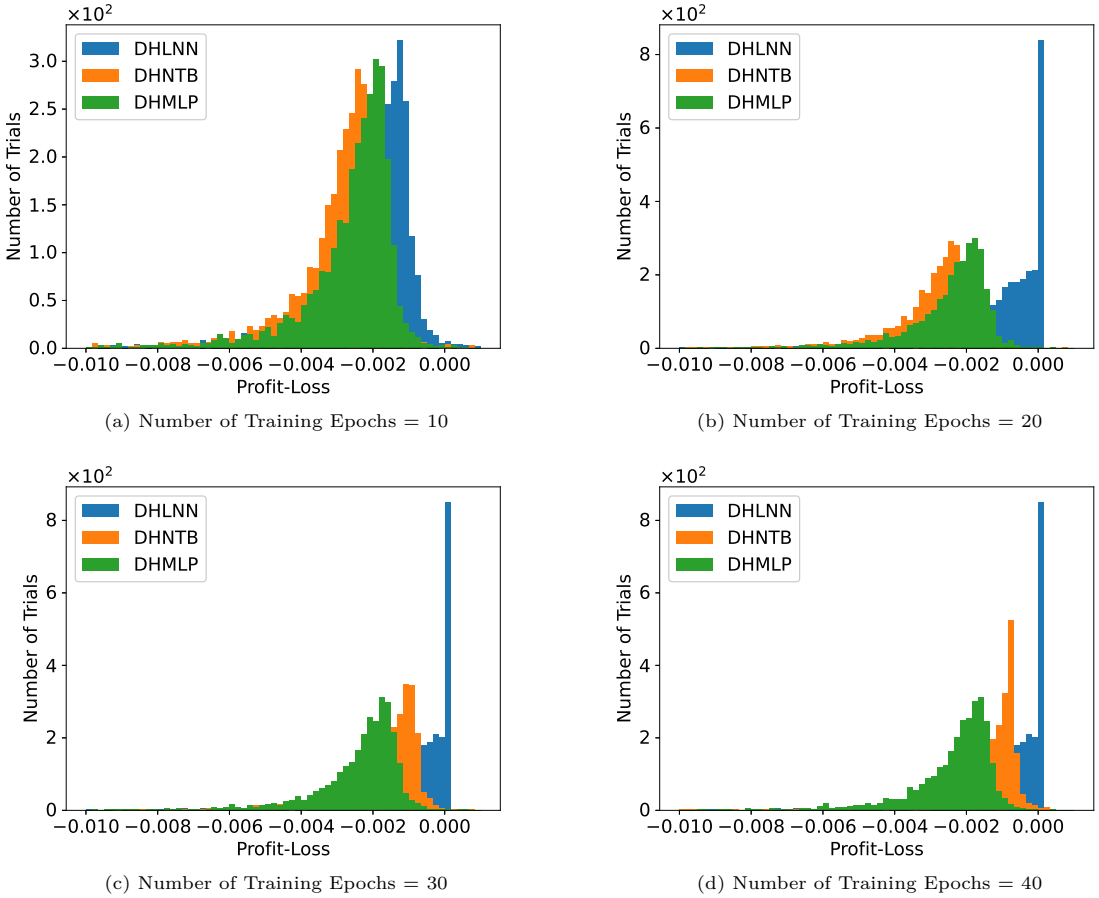


Figure 10: Comparison of hedging performance across various transaction costs $\{2 \times 10^{-3}, 4 \times 10^{-3}, 6 \times 10^{-3}, 8 \times 10^{-3}\}$ for a Lookback option with a strike price of 1.0. The performance is evaluated over 50 training epochs, focusing on the Expected Shortfall of the hedging PNL.



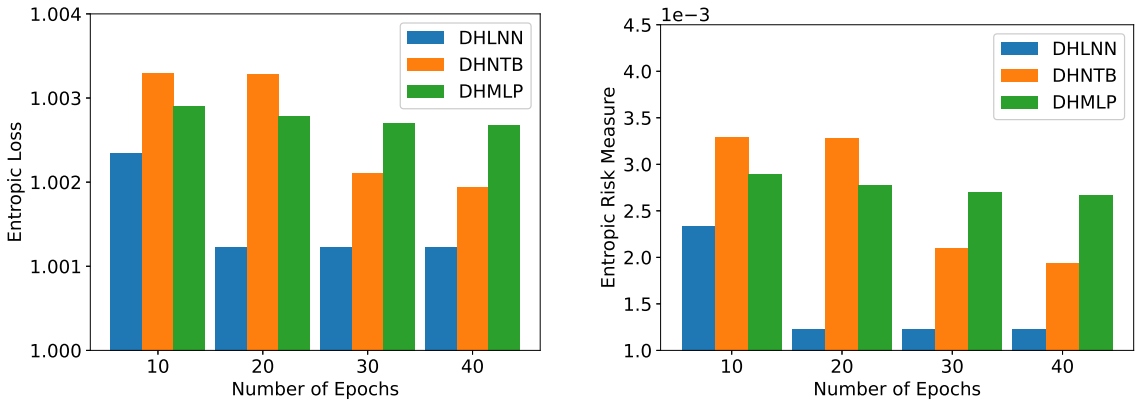
(a) Number of Training Epochs = 10

(b) Number of Training Epochs = 20

(c) Number of Training Epochs = 30

(d) Number of Training Epochs = 40

Figure 11: Comparison of convergence performance across different training epochs $\{10, 20, 30, 40\}$ for deep hedging models applied to a Lookback option with a strike price of 1.0. The analysis considers a transaction cost of 2×10^{-3}.



(a) (b)

Figure 12: Convergence Performance comparison over different training epochs $\{10, 20, 30, 40\}$ for deep hedging models over a Lookback option with strike 1.0 for Entropic Loss and Entropic Risk Measure of hedging PNL, transaction cost is 2×10^{-3}.

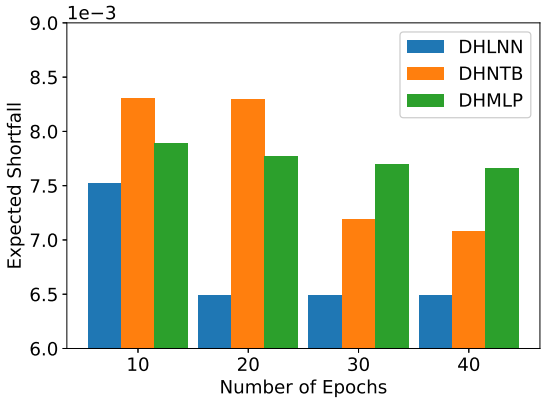


Figure 13: Convergence performance comparison of deep hedging models over different training epochs $\{10, 20, 30, 40\}$ for a Lookback option with a strike price of 1.0. The analysis focuses on the Expected Shortfall of the hedging PNL with a transaction cost of 2×10^{-3}.

superior risk management and more effective hedging strategies, even with fewer training epochs. This efficiency highlights DHLNN's robustness in managing Lookback options.

Fig. 12 compares Entropic Loss and Entropic Risk Measure for the same models and training settings. DHLNN consistently outperforms DHMLP and DHNTB, with noticeable improvements evident by 20 epochs. The rapid convergence of DHLNN, as shown in the low and stable metrics at 20 epochs, demonstrates its efficiency in learning optimal hedging strategies with minimal training efforts. Beyond 20 epochs, DHLNN's performance stabilizes, indicating that it has already converged to its optimal state. While DHNTB shows improvements with extended training, it requires more epochs to surpass DHMLP, which initially manages risk more effectively. The results show DHLNN's ability to achieve superior risk management and hedging performance with significantly fewer training epochs compared to DHMLP and DHNTB. DHLNN's rapid convergence and consistent performance across all metrics highlight its advantage in delivering reliable and efficient hedging strategies for complex derivatives like Lookback options.

Fig. 13 analyzes Expected Shortfall across the same models and training epochs. Initially, DHMLP outperforms DHNTB in managing tail risk indicated by lower Expected Shortfall values. However, as training progresses, DHNTB surpasses DHMLP, suggesting that it requires more training to optimize its strategies effectively. In contrast, DHLNN consistently achieves the lowest Expected Shortfall values, with significant improvements evident by 20 epochs. The rapid stabilization of DHLNN's Expected Shortfall values indicates its efficiency in learning robust hedging strategies with minimal training. Beyond 20 epochs, further improvements are

marginal, demonstrating DHLNN's quick convergence to an optimal state. The results highlight DHLNN's resilience and reliability in managing extreme market movements, especially when training resources are limited.

7 Conclusion

In conclusion, our work introduces a cutting-edge linearized neural network architecture that seamlessly integrates with the Black-Scholes delta, offering significant potential to reduce transaction costs and establish robust hedge positions in derivative markets. This approach enhances both the robustness and interpretability of hedging strategies while preserving the powerful representational capabilities inherent in neural networks. By framing the model training procedure within the context of convex risk measures, we have established a transparent and mathematically sound framework for risk assessment and the optimization of hedging strategies. One critical future research area is the development of robust, explainable, and adaptive deep hedging strategies that can dynamically adjust to evolving market conditions. As financial markets continue to become more complex and interconnected, the need for hedging strategies that can provide both robustness against market volatility and clarity in their decision-making processes becomes increasingly important. Enhancing the adaptive capabilities of these models to offer more responsive and dynamic hedging strategies remains an open challenge, and addressing this will be key to advancing the practical applicability of deep hedging techniques in real-world scenarios.

References

CameronOptiver IXAGPOPU Jiashen Liu Matteo Pietrobon (Optiver) OptiverMerle Sohier Dane Stefan Vallentine Andrew Meyer, BerniceOptiver. Optiver realized volatility prediction, 2021. URL `https://kaggle.com/competitions/optiver-realized-volatility-prediction`.

Aharon Ben-Tal and Marc Teboulle. An old-new concept of convex risk measures: The optimized certainty equivalent. *Mathematical Finance*, 17(3):449–476, 2007.

Fischer Black and Myron Scholes. The pricing of options and corporate liabilities. *Journal of political economy*, 81(3):637–654, 1973.

Mark Broadie, Paul Glasserman, and Shing-Gang Kou. Connecting discrete and continuous path-dependent options. *Finance and Stochastics*, 3:55–82, 1999.

Hans Buehler, Lukas Gonon, Josef Teichmann, and Ben Wood. Deep hedging. *Quantitative Finance*, 19(8): 1271–1291, 2019.

Jay Cao, Jacky Chen, and John Hull. A neural network approach to understanding implied volatility movements. *Quantitative Finance*, 20(9):1405–1413, 2020.

René Carmona. *Indifference pricing: theory and applications*. Princeton University Press, 2008.

Bruce M Collins and Frank J Fabozzi. A methodology for measuring transaction costs. *Financial Analysts Journal*, 47(2):27–36, 1991.

Shom Prasad Das and Sudarsan Padhy. A new hybrid parametric and machine learning model with homogeneity hint for european-style index option pricing. *Neural Computing and Applications*, 28:4061–4077, 2017.

Mark HA Davis, Vassilios G Panas, and Thaleia Zariphopoulou. European option pricing with transaction costs. *SIAM Journal on Control and Optimization*, 31(2):470–493, 1993.

Erik R Grannan and Glen H Swindle. Minimizing transaction costs of option hedging strategies. *Mathematical finance*, 6(4):341–364, 1996.

Vicky Henderson and David Hobson. Utility indifference pricing-an overview. *Volume on Indifference Pricing*, 2004.

John Hull and Alan White. The pricing of options on assets with stochastic volatilities. *The journal of finance*, 42(2):281–300, 1987.

John C Hull and Sankarshan Basu. *Options, futures, and other derivatives*. Pearson Education India, 2016.

Kentaro Imajo, Kentaro Minami, Katsuya Ito, and Kei Nakagawa. Deep portfolio optimization via distributional prediction of residual factors. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 213–222, 2021.

Shota Imaki, Kentaro Imajo, Katsuya Ito, Kentaro Minami, and Kei Nakagawa. No-transaction band network: A neural network architecture for efficient deep hedging. *arXiv preprint arXiv:2103.01775*, 2021.

Huisu Jang and Jaewook Lee. Generative bayesian neural network model for risk-neutral pricing of american index options. *Quantitative Finance*, 19(4):587–603, 2019.

Jan Kallsen and Johannes Muhle-Karbe. Option pricing and hedging with small transaction costs. *Mathematical Finance*, 25(4):702–723, 2015.

Ryo Karakida, Shotaro Akaho, and Shun-ichi Amari. Universal statistics of fisher information in deep neural networks: Mean field approach. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1032–1041. PMLR, 2019.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

Hayne E Leland. Option pricing and replication with transactions costs. *The journal of finance*, 40(5): 1283–1301, 1985.

Song Mei, Andrea Montanari, and Phan-Minh Nguyen. A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33):E7665–E7671, 2018.

Michael Monoyios. Option pricing with transaction costs using a markov chain approximation. *Journal of Economic Dynamics and Control*, 28(5):889–913, 2004.

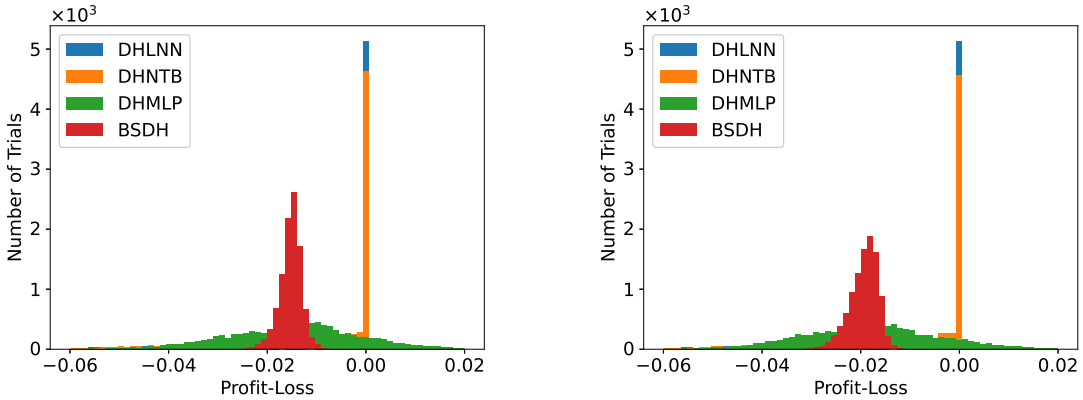
Roman Novak, Jascha Sohl-Dickstein, and Samuel S Schoenholz. Fast finite width neural tangent kernel. In *International Conference on Machine Learning*, pp. 17018–17044. PMLR, 2022.

Johannes Ruf and Weiguan Wang. Neural networks for option pricing and hedging: a literature review. *arXiv preprint arXiv:1911.05620*, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

A Performance Comparison with Additional Simulations

A.1 Simulations for Lookback Options



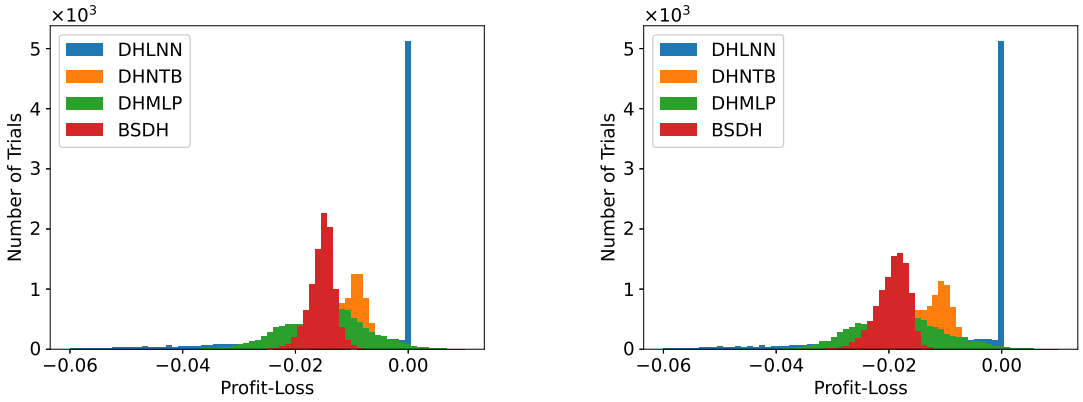
(a) Cost $= 2 \times 10^{-3}$

(b) Cost $= 4 \times 10^{-3}$

(c) Cost $= 6 \times 10^{-3}$

(d) Cost $= 8 \times 10^{-3}$

Figure 14: Comparison of hedging performance across different transaction costs $\{2 \times 10^{-3}, 4 \times 10^{-3}, 6 \times 10^{-3}, 8 \times 10^{-3}\}$ for a European option with a strike price of 1.0. The analysis, conducted over 100 training epochs, focuses on the distribution of hedging PNL with volatility set at 0.1.



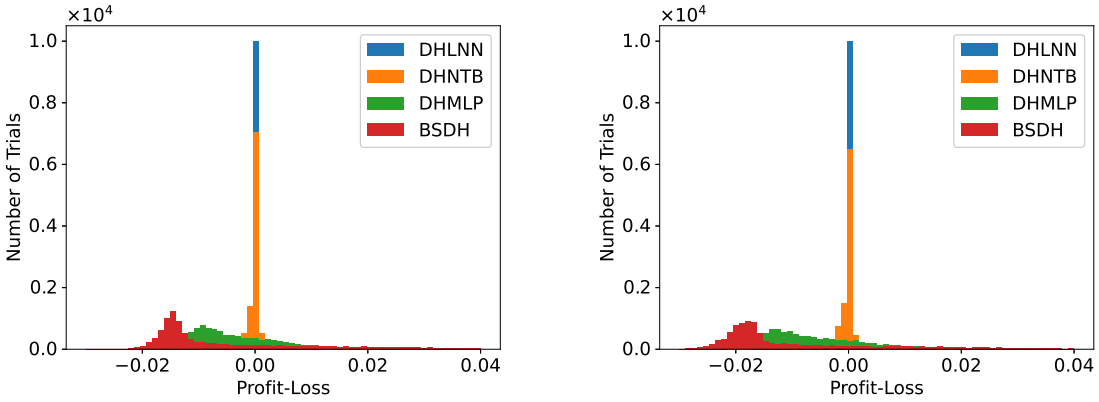
(a) Cost $= 2 \times 10^{-3}$

(b) Cost $= 4 \times 10^{-3}$

(c) Cost $= 6 \times 10^{-3}$

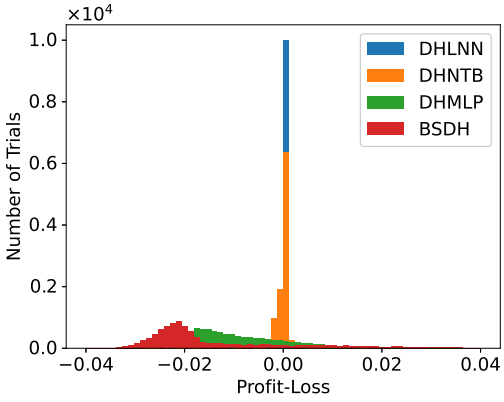
(d) Cost $= 8 \times 10^{-3}$

Figure 15: Comparison of hedging performance across different transaction costs $\{2 \times 10^{-3}, 4 \times 10^{-3}, 6 \times 10^{-3}, 8 \times 10^{-3}\}$ for a European option with a strike price of 1.0. The analysis, conducted over 50 training epochs, focuses on the distribution of hedging PNL with volatility set at 0.1.

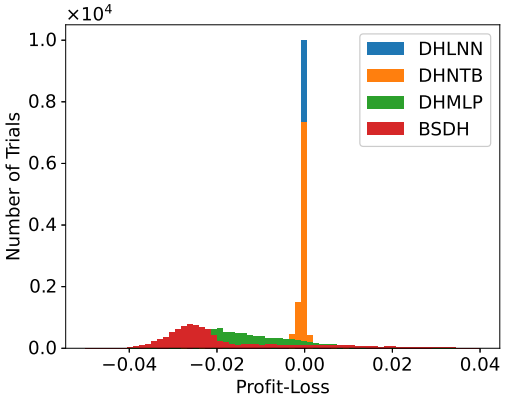


(a) Cost $= 2 \times 10^{-3}$

(b) Cost $= 4 \times 10^{-3}$

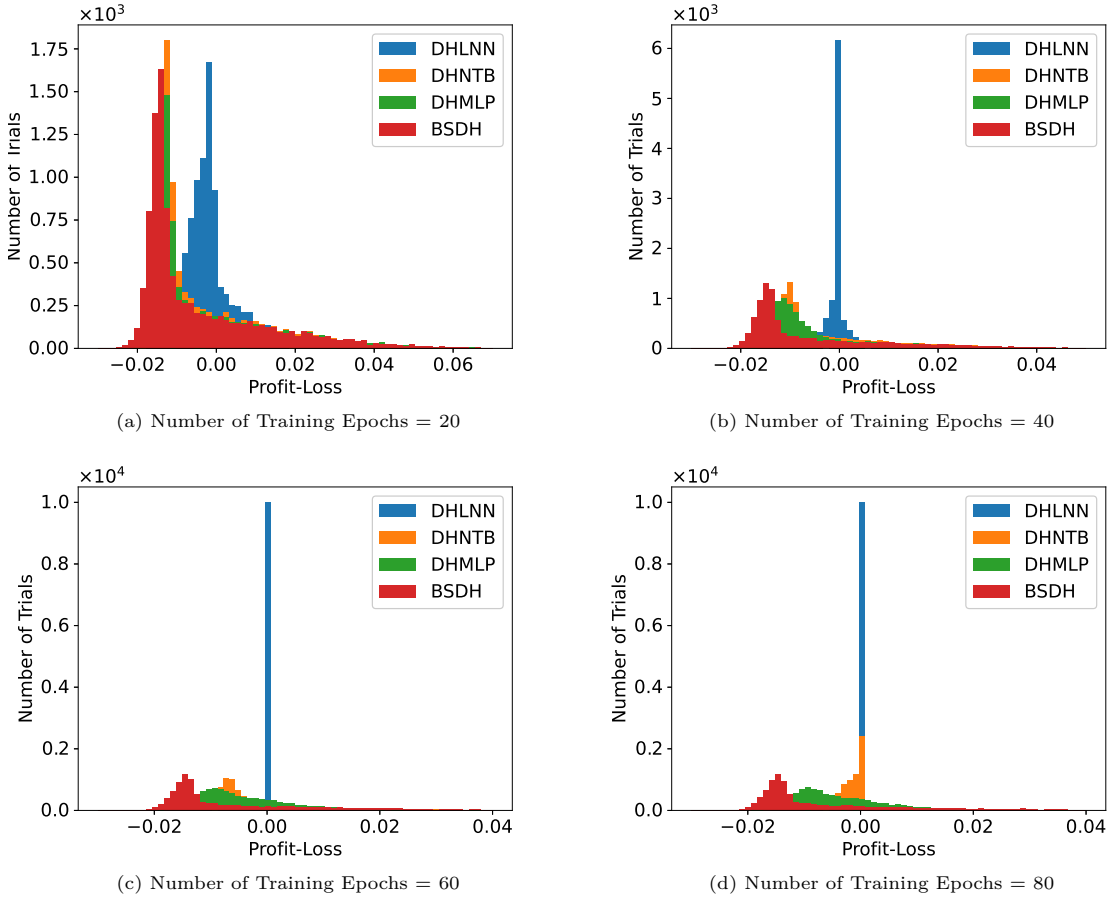


(c) Cost $= 6 \times 10^{-3}$



(d) Cost $= 8 \times 10^{-3}$

Figure 16: Comparison of hedging performance across different transaction costs $\{2 \times 10^{-3}, 4 \times 10^{-3}, 6 \times 10^{-3}, 8 \times 10^{-3}\}$ for a European option with a strike price of 1.2. The analysis, conducted over 100 training epochs, focuses on the distribution of hedging PNL with volatility set at 0.1.



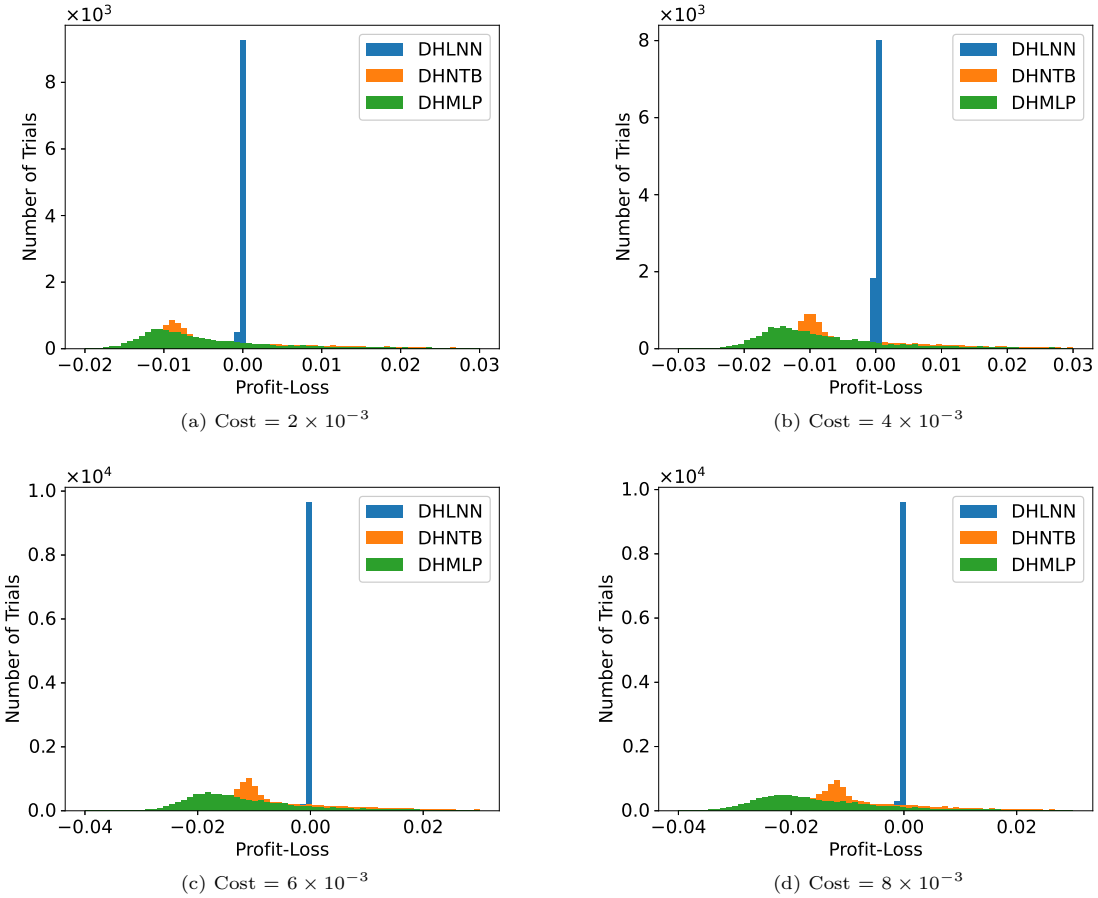
(a) Number of Training Epochs = 20

(b) Number of Training Epochs = 40

(c) Number of Training Epochs = 60

(d) Number of Training Epochs = 80

Figure 17: Comparison of convergence performance for deep hedging models across different training epochs $\{20, 40, 60, 80\}$ on a European option with a strike price of 1.2. The analysis includes transaction costs of 2×10^{-3} and a fixed volatility of 0.1.



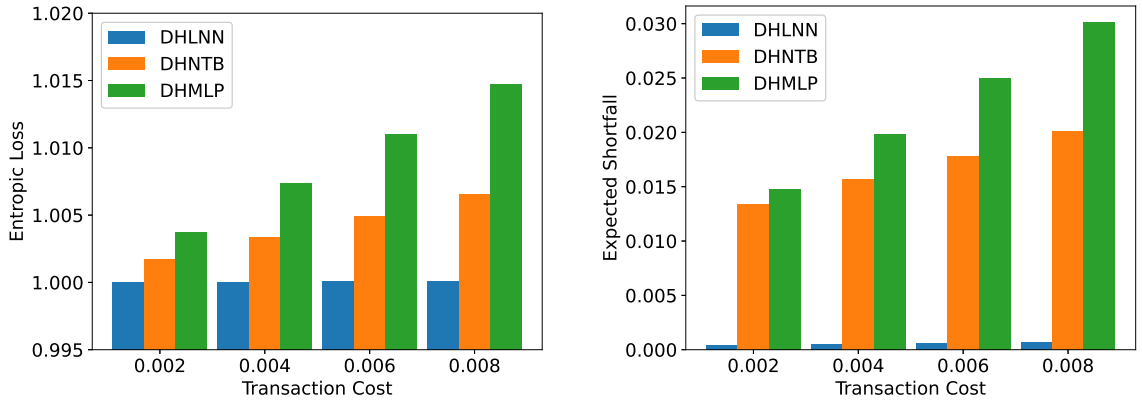
(a) Cost $= 2 \times 10^{-3}$

(b) Cost $= 4 \times 10^{-3}$

(c) Cost $= 6 \times 10^{-3}$

(d) Cost $= 8 \times 10^{-3}$

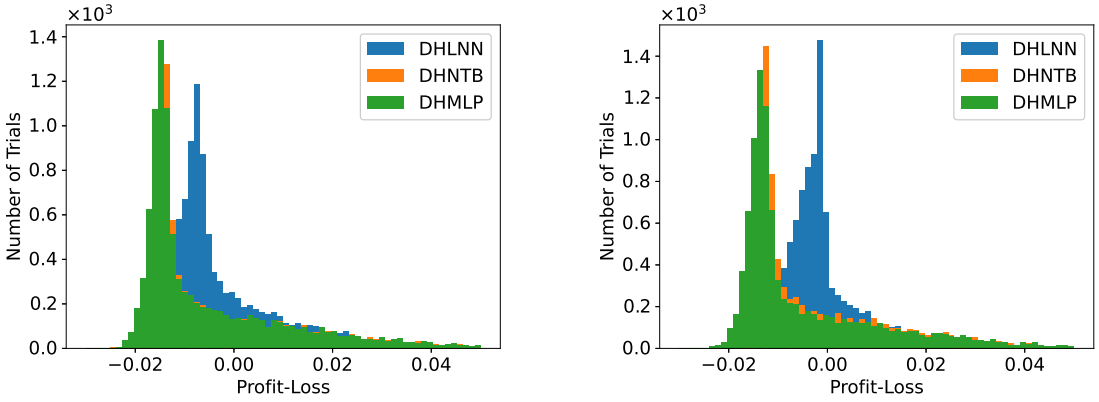
Figure 18: Hedging Performance comparison over different transaction costs $\{2 \times 10^{-3}, 4 \times 10^{-3}, 6 \times 10^{-3}, 8 \times 10^{-3}\}$ of underlying asset for a Lookback option and strike 1.2 with 50 training epochs for the distribution of Hedging PNL where volatility is set as 0.1.



(a)

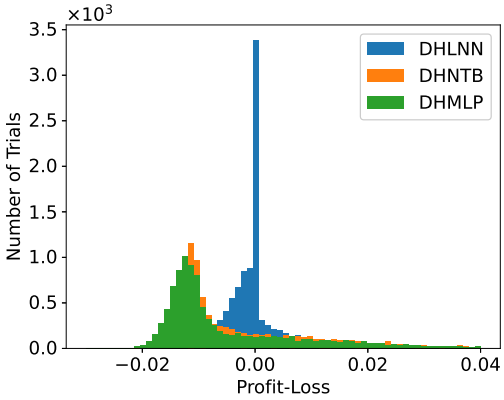
(b)

Figure 19: Hedging Performance comparison over different transaction costs $\{2 \times 10^{-3}, 4 \times 10^{-3}, 6 \times 10^{-3}, 8 \times 10^{-3}\}$ of underlying asset for a Lookback option and strike 1.2 with 50 training epochs for the Entropic Loss and Expected Shortfall of Hedging PNL where volatility is set as 0.1.

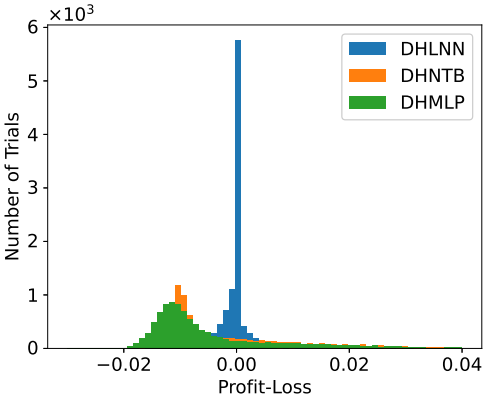


(a) Number of Training Epochs = 10

(b) Number of Training Epochs = 20

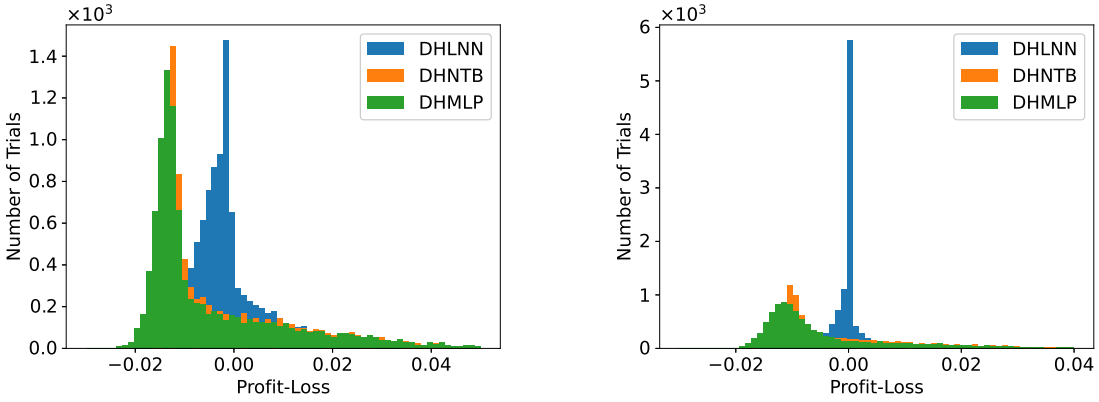


(c) Number of Training Epochs = 30



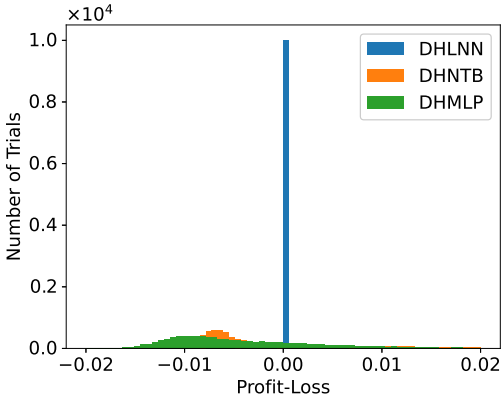
(d) Number of Training Epochs = 40

Figure 20: Comparison of convergence performance across different training epochs $\{10, 20, 30, 40\}$ for deep hedging models applied to a Lookback option with a strike price of 1.2. The analysis considers a transaction cost of 2×10^{-3}.

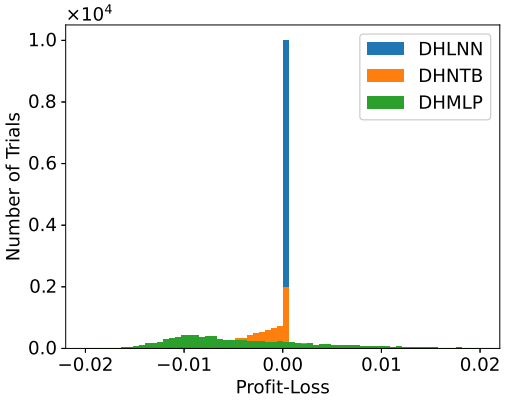


(a) Number of Training Epochs = 20

(b) Number of Training Epochs = 40

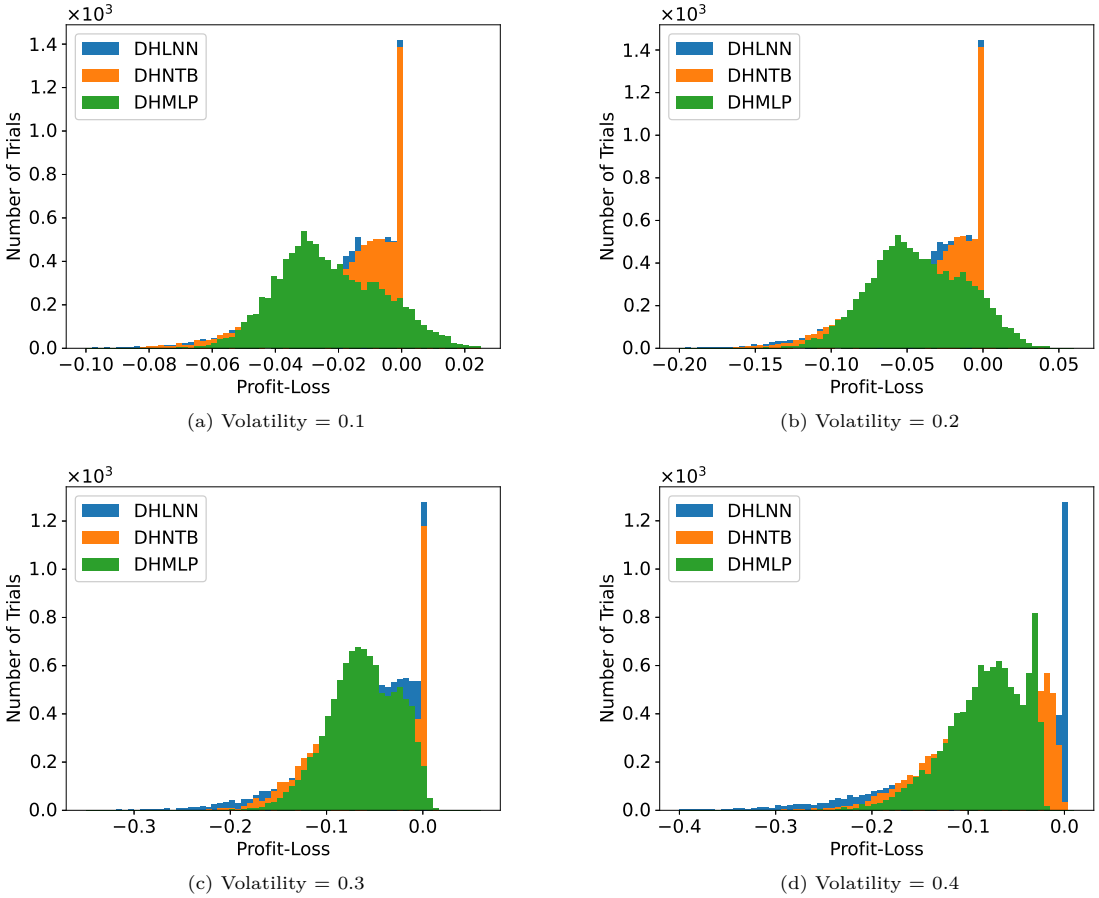


(c) Number of Training Epochs = 60



(d) Number of Training Epochs = 80

Figure 21: Comparison of convergence performance across different training epochs $\{10, 20, 30, 40\}$ for deep hedging models applied to a Lookback option with a strike price of 1.2. The analysis considers a transaction cost of 2×10^{-3}.



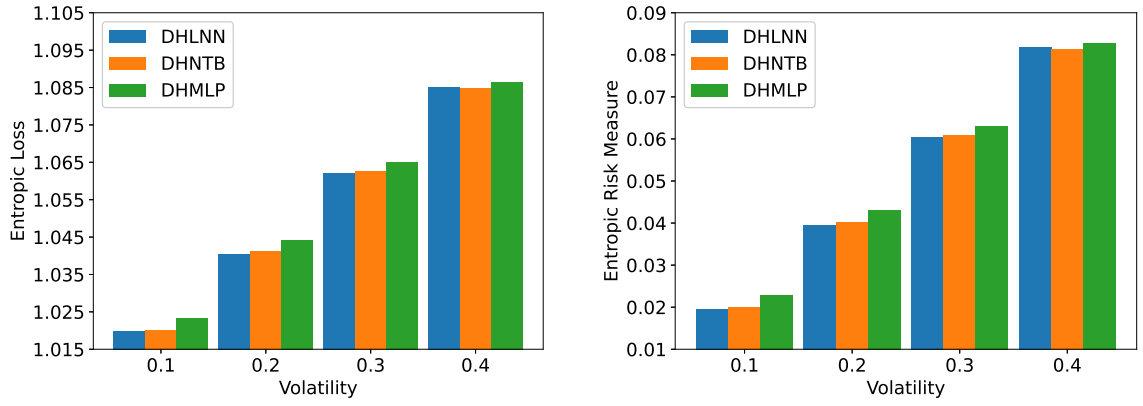
(a) Volatility = 0.1

(b) Volatility = 0.2

(c) Volatility = 0.3

(d) Volatility = 0.4

Figure 22: Hedging Performance comparison of PNL distributions over different volatility of the underline asset price $\{0.1, 0.2, 0.3, 0.4\}$ for a Lookback option and strike 1.0 with transaction cost 2×10^{-3}, with 100 training epochs.



(a)

(b)

Figure 23: Hedging Performance comparison of PNL distributions over different volatility of the underline asset price $\{0.1, 0.2, 0.3, 0.4\}$ for a Lookback option and strike 1.0 for Entropic Loss and Entropic Risk Measure of hedging PNL, transaction cost 2×10^{-3}, with 100 training epochs.